

## CASTRO: A NEW COMPRESSIBLE ASTROPHYSICAL SOLVER. II. GRAY RADIATION HYDRODYNAMICS

W. ZHANG<sup>1</sup>, L. HOWELL<sup>2</sup>, A. ALMGREN<sup>1</sup>, A. BURROWS<sup>3</sup>, J. BELL<sup>1</sup>*Draft version January 20, 2013*

## ABSTRACT

We describe the development of a flux-limited gray radiation solver for the compressible astrophysics code, CASTRO. CASTRO uses an Eulerian grid with block-structured adaptive mesh refinement based on a nested hierarchy of logically-rectangular variable-sized grids with simultaneous refinement in both space and time. The gray radiation solver is based on a mixed-frame formulation of radiation hydrodynamics. In our approach, the system is split into two parts, one part that couples the radiation and fluid in a hyperbolic subsystem, and another parabolic part that evolves radiation diffusion and source-sink terms. The hyperbolic subsystem is solved explicitly with a high-order Godunov scheme, whereas the parabolic part is solved implicitly with a first-order backward Euler method.

*Subject headings:* diffusion – hydrodynamics – methods: numerical – radiative transfer

## 1. INTRODUCTION

In this paper, we present the development of a gray radiation solver in our compressible astrophysics code, CASTRO. CASTRO uses an Eulerian grid with block-structured adaptive mesh refinement (AMR). Our approach to AMR is based on a nested hierarchy of logically-rectangular variable-sized grids with simultaneous refinement in both space and time. In our previous paper (Almgren et al. 2010, henceforth Paper I), we describe our treatment of hydrodynamics, including gravity and nuclear reactions. Here, we describe an algorithm for flux-limited gray radiation hydrodynamics based on a mixed-frame formulation.

Many astrophysical phenomena involve radiative processes, which often dominate the energy transport and dynamical behavior of the system. Some examples include star formation, stellar structure and evolution, accretion onto compact objects, and supernovae. Radiation hydrodynamics simulations are playing an increasingly important role in modeling these astrophysical systems.

The fundamental equation of radiation transfer is a six-dimensional integro-differential equation (Pomraning 1973), which is unfortunately very difficult to solve. Numerical codes typically solve one or two angular moment equations of the transfer equation. One common approach is to solve a two-moment system including radiation energy density and radiation flux (e.g., Hayes & Norman 2003; Hubeny & Burrows 2007; González et al. 2007; Sekora & Stone 2010). The system is closed by an approximate expression for the radiation pressure. Another popular approach is to solve the radiation energy equation only (e.g., Turner & Stone 2001; Hayes et al. 2006; Krumholz et al. 2007; Swesty & Myra 2009; Commerçon et al. 2011; van der Holst et al. 2011). In this approach, the so-called flux-limited diffusion (FLD) approximation is used for closure (Alme & Wilson 1973). The two-moment approach is more accurate when

the radiation is highly anisotropic and optically-thin, but it is computationally more expensive than the FLD approach. However, FLD is a very good approximation for optically-thick flows. Furthermore, the FLD approach can be numerically more robust than the two-moment approach for systems in the hyperbolic limit. We have adopted the FLD approach in the radiation solver of CASTRO.

CASTRO solves the equations of nonrelativistic radiation hydrodynamics. Thus, gas quantities, such as pressure, temperature and density, are treated as frame-independent because the corrections are of order  $O(v^2/c^2)$ , where  $v$  is the gas velocity and  $c$  is the speed of light. However, radiation quantities, such as radiation energy density, radiation flux, and radiation pressure, in the comoving frame differ from those in the laboratory frame by order  $O(v/c)$  (Mihalas & Mihalas 1999). Neglecting the  $O(v/c)$  terms potentially leads to erroneous results, especially in the dynamic diffusion limit where transport of radiation is dominated by motion of the fluid (Castor 1972; Mihalas & Klein 1982; Castor 2004). For numerical codes, some authors chose the comoving frame approach in which the radiation quantities are measured in the comoving frame (e.g., Turner & Stone 2001; Hayes & Norman 2003; González et al. 2007; Swesty & Myra 2009; Commerçon et al. 2011), whereas others chose the mixed-frame approach in which the radiation quantities are computed in the laboratory frame while the opacities are measured in the comoving frame (e.g., Mihalas & Klein 1982; Hubeny & Burrows 2007; Krumholz et al. 2007; Sekora & Stone 2010). A primary weakness of the comoving frame approach is that it does not conserve energy, whereas the mixed-frame approach is not suitable for systems in which line transport is important. In CASTRO, we have chosen the mixed-frame approach because it conserves the total energy and is well suited for AMR.

This paper is organized as follows. In § 2 we present the governing equations of the mixed-frame gray radiation hydrodynamics and the mathematical characteristics of the system. In § 3 we describe the single-level integration scheme. In § 4 we describe how the integration algorithm is extended for AMR. In § 5 we show the scaling behavior

<sup>1</sup> Center for Computational Sciences and Engineering, Lawrence Berkeley National Laboratory, Berkeley, CA 94720

<sup>2</sup> Center for Applied Scientific Computing, Lawrence Livermore National Laboratory, Livermore, CA 94550

<sup>3</sup> Dept. of Astrophysical Sciences, Princeton University, Princeton, NJ 08544

of CASTRO with radiation. In § 6 we present results from a series of test problems. Finally, we summarize the results of the paper in § 7.

## 2. GRAY RADIATION HYDRODYNAMICS

### 2.1. Equations of Gray Radiation Hydrodynamics

Assuming local thermodynamic equilibrium, the mixed-frame frequency-integrated radiation hydrodynamics equations, correct to  $O(v/c)$ , can be written as (see e.g., Mihalas & Klein 1982; Lowrie et al. 1999):

$$\frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \mathbf{u}) = 0, \quad (1)$$

$$\frac{\partial(\rho \mathbf{u})}{\partial t} + \nabla \cdot (\rho \mathbf{u} \mathbf{u}) + \nabla p = \frac{1}{c} \chi_F \mathbf{F}_r^{(0)} - \kappa_P \left( \frac{\mathbf{u}}{c} \right) (aT^4 - E_r^{(0)}), \quad (2)$$

$$\frac{\partial(\rho E)}{\partial t} + \nabla \cdot (\rho E \mathbf{u} + p \mathbf{u}) = -c\kappa_P (aT^4 - E_r^{(0)}) + \chi_F \left( \frac{\mathbf{u}}{c} \right) \cdot \mathbf{F}_r^{(0)}, \quad (3)$$

$$\frac{\partial E_r}{\partial t} + \nabla \cdot \mathbf{F}_r = c\kappa_P (aT^4 - E_r^{(0)}) - \chi_F \left( \frac{\mathbf{u}}{c} \right) \cdot \mathbf{F}_r^{(0)}, \quad (4)$$

$$\frac{1}{c^2} \frac{\partial \mathbf{F}_r}{\partial t} + \nabla \cdot \mathbf{P}_r = -\frac{1}{c} \chi_F \mathbf{F}_r^{(0)} + \kappa_P \left( \frac{\mathbf{u}}{c} \right) (aT^4 - E_r^{(0)}). \quad (5)$$

Here  $\rho$ ,  $\mathbf{u}$ ,  $p$ ,  $T$ , and  $E$  are the mass density, velocity, pressure, temperature, and total energy per unit mass (internal energy,  $e$ , plus kinetic energy,  $u^2/2$ ), respectively.  $E_r$ ,  $\mathbf{F}_r$ , and  $\mathbf{P}_r$  are radiation energy density, radiation flux, and radiation pressure tensor, respectively. Note that here the  $r$  subscript denotes radiation. The speed of light and radiation constant are denoted by  $c$  and  $a$ , respectively, where  $a = 4\sigma/c$  and  $\sigma$  is the Stefan-Boltzmann constant.  $\kappa_P$  and  $\chi_F$  are the Planck mean and flux mean interaction coefficients, both in units of inverse length. The (0) superscript denotes the comoving frame. Radiation quantities ( $E_r$ ,  $\mathbf{F}_r$  and  $\mathbf{P}_r$ ) without the (0) superscript are measured in the lab frame. Radiation quantities measured in the comoving and lab frames are related by the Lorentz transformation (Mihalas & Klein 1982). It should be noted that absorption and scattering coefficients are always computed in the comoving frame in the mixed-frame approach. Also note that scattering can be included in the flux mean interaction coefficient. The whole system is closed by an equation of state for the fluid and a relation between  $\mathbf{P}_r^{(0)}$  and  $E_r^{(0)}$ ,

$$\mathbf{P}_r^{(0)} = \mathbf{f}^{(0)} E_r^{(0)}, \quad (6)$$

where  $\mathbf{f}^{(0)}$  is the Eddington tensor in the comoving frame.

In the FLD approximation (Alme & Wilson 1973), the comoving radiation flux is written in the form of Fick's law of diffusion,

$$\mathbf{F}_r^{(0)} = -D \nabla E_r^{(0)}, \quad (7)$$

where the diffusion coefficient  $D$  is given by

$$D = \frac{c\lambda}{\chi_R}, \quad (8)$$

where  $\chi_R$  is the Rosseland mean of the sum of the absorption and scattering coefficients, and  $\lambda$  is the flux limiter. We adopt the flux limiter approximation given in Levermore & Pomraning (1981) as

$$\lambda = \frac{2 + R}{6 + 3R + R^2}, \quad (9)$$

$$R = \frac{|\nabla E_r^{(0)}|}{\chi_R E_r^{(0)}}. \quad (10)$$

The corresponding radiation pressure tensor is (Levermore 1984)

$$\mathbf{P}_r^{(0)} = \frac{1}{2}[(1-f)\mathbf{I} + (3f-1)\hat{\mathbf{n}}\hat{\mathbf{n}}]E_r^{(0)}, \quad (11)$$

where  $\mathbf{I}$  is the identity tensor of rank 2,  $\hat{\mathbf{n}} = \nabla E_r^{(0)} / |\nabla E_r^{(0)}|$ , and the Eddington factor  $f$  is given by

$$f = \lambda + \lambda^2 R^2. \quad (12)$$

We note that in the optically-thick limit both the flux limiter  $\lambda$  and the Eddington factor  $f$  approach  $1/3$ , whereas in the optically-thin limit the flux limiter  $\lambda$  and the Eddington factor  $f$  approach 0 and 1, respectively.

Furthermore we assume that  $\chi_F = \chi_R$ , a common approximation accurate in optically-thick regions (Mihalas & Mihalas 1999). Following Krumholz et al. (2007), we keep terms up to  $O(v/c)$ , and we drop all terms that are insignificant in all following regimes: streaming, static diffusion, and dynamic diffusion limits. Our radiation hydrodynamics equations now become,

$$\frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \mathbf{u}) = 0, \quad (13)$$

$$\frac{\partial(\rho \mathbf{u})}{\partial t} + \nabla \cdot (\rho \mathbf{u} \mathbf{u}) + \nabla p + \lambda \nabla E_r = 0, \quad (14)$$

$$\frac{\partial(\rho E)}{\partial t} + \nabla \cdot (\rho E \mathbf{u} + p \mathbf{u}) + \lambda \mathbf{u} \cdot \nabla E_r = -c\kappa_P (aT^4 - E_r^{(0)}), \quad (15)$$

$$\frac{\partial E_r}{\partial t} + \nabla \cdot \left( \frac{3-f}{2} E_r \mathbf{u} \right) - \lambda \mathbf{u} \cdot \nabla E_r = c\kappa_P (aT^4 - E_r^{(0)}) + \nabla \cdot \left( \frac{c\lambda}{\chi_R} \nabla E_r \right). \quad (16)$$

The absorption terms on the right hand side of these equations still include the radiation energy density in the comoving frame, because this is the frame in which emission and absorption balance as the material becomes opaque. The comoving and lab frame quantities are related by

$$E_r^{(0)} = E_r - \frac{2}{c^2} \mathbf{u} \cdot \mathbf{F}_r^{(0)} + O(v^2/c^2) \quad (17)$$

$$= E_r + 2 \frac{\lambda}{\chi_R} \frac{\mathbf{u}}{c} \cdot \nabla E_r + O(v^2/c^2) \quad (18)$$

in the framework of the FLD approximation.

### 2.2. Mathematical Characteristics of the Hyperbolic Subsystem

Radiation hydrodynamics under the assumption of FLD is a mixed hyperbolic-parabolic system with stiff

source terms. The equations of the hyperbolic subsystem are

$$\frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \mathbf{u}) = 0, \quad (19)$$

$$\frac{\partial(\rho \mathbf{u})}{\partial t} + \nabla \cdot (\rho \mathbf{u} \mathbf{u}) + \nabla p + \lambda \nabla E_r = 0, \quad (20)$$

$$\frac{\partial(\rho E)}{\partial t} + \nabla \cdot (\rho E \mathbf{u} + p \mathbf{u}) + \lambda \mathbf{u} \cdot \nabla E_r = 0, \quad (21)$$

$$\frac{\partial E_r}{\partial t} + \nabla \cdot \left( \frac{3-f}{2} E_r \mathbf{u} \right) - \lambda \mathbf{u} \cdot \nabla E_r = 0, \quad (22)$$

which are obtained by neglecting the terms on the right-hand-side of Eqs. 13–16. In the limit of strong equilibrium (i.e.,  $E_r^{(0)} \approx aT^4$  and  $\chi_R \rightarrow \infty$ ), these right-hand-side terms are negligible and the full system becomes hyperbolic, governed by Eqs. 19–22. In the more general case the hyperbolic subsystem will be solved first as part of a time-split discretization.

In CASTRO, we solve the hyperbolic subsystem with a Godunov method, which utilizes a characteristic-based Riemann solver. The Godunov method requires that we analyze the mathematical characteristics of the hyperbolic subsystem. For simplicity, let us consider the system in one dimension, which can be written in terms of primitive variables as,

$$\frac{\partial Q}{\partial t} + A \frac{\partial Q}{\partial x} = 0, \quad (23)$$

where the primitive variables are

$$Q = \begin{pmatrix} \rho \\ u \\ p \\ E_r \end{pmatrix}, \quad (24)$$

and the Jacobian matrix is

$$A = \begin{pmatrix} u & \rho & 0 & 0 \\ 0 & u & \frac{1}{\rho} & \frac{\lambda}{\rho} \\ 0 & \gamma p & u & 0 \\ 0 & \frac{3-f}{2} E_r & 0 & (\frac{3-f}{2} - \lambda) u \end{pmatrix}. \quad (25)$$

The system is hyperbolic because the Jacobian matrix is diagonalizable with four real eigenvalues. In general cases, the eigenvalues and eigenvectors are unfortunately very complicated. However, when the following relation holds,

$$\frac{3-f}{2} = \lambda + 1, \quad (26)$$

the four eigenvalues are,

$$u - c_s, \quad u, \quad u, \quad u + c_s, \quad (27)$$

where

$$c_s = \sqrt{\gamma \frac{p}{\rho} + (\lambda + 1) \frac{\lambda E_r}{\rho}} \quad (28)$$

is the radiation modified sound speed. The correspond-

ing right eigenvectors are,

$$\begin{pmatrix} 1 \\ -c_s/\rho \\ \gamma p/\rho \\ (\lambda + 1)E_r/\rho \end{pmatrix}, \begin{pmatrix} 0 \\ 0 \\ -\lambda \\ 1 \end{pmatrix}, \begin{pmatrix} 1 \\ 0 \\ 0 \\ 0 \end{pmatrix}, \begin{pmatrix} 1 \\ c_s/\rho \\ \gamma p/\rho \\ (\lambda + 1)E_r/\rho \end{pmatrix}, \quad (29)$$

and the corresponding left eigenvectors are,

$$\begin{pmatrix} 0, & -\rho/2c_s, & 1/2c_s^2, & \lambda/2c_s^2, \\ 0, & 0, & -(\lambda + 1)E_r/\rho c_s^2, & \gamma p/\rho c_s^2, \\ 1, & 0, & -1/c_s^2, & -\lambda/c_s^2, \\ 0, & \rho/2c_s, & 1/2c_s^2, & \lambda/2c_s^2. \end{pmatrix} \quad (30)$$

These four eigenvectors define the characteristic fields for the one-dimensional system. By computing the product of the right eigenvectors and the gradients of their corresponding eigenvalues (LeVeque 2002), we find that the first and fourth fields are genuinely nonlinear corresponding to either a shock wave or a rarefaction wave. The second and third fields are linearly degenerate corresponding to a contact discontinuity in either gas pressure or density. Note that there is also a jump in radiation energy density accompanying the jump in gas pressure such that the total pressure,  $p_{\text{tot}} = p + \lambda E_r$ , is constant across the contact discontinuity. Obviously, in three dimensions, there are two additional linear discontinuities for transverse velocities just like the case of pure hydrodynamics.

It should be noted that Eq. 26 is satisfied in both optically-thick and thin limits. Although this condition is not always satisfied, it is a fairly good approximation. The Levermore & Pomraning (1981) flux limiter that we use (Eqs. 9 & 10) satisfies

$$0.978 < \frac{3-f}{2} - \lambda \leq 1.05. \quad (31)$$

Thus, for the purpose of an approximate Riemann solver, the approximate eigenvalues and eigenvectors are used.

### 2.3. Radiation Diffusion and Source-Sink Terms

The parabolic part of the system consists of the radiation diffusion and source-sink terms, which were omitted from the discussion of the hyperbolic subsystem (Eqs. 19–22).

$$\frac{\partial(\rho e)}{\partial t} = -c\kappa_P(aT^4 - E_r^{(0)}), \quad (32)$$

$$= -c\kappa_P(aT^4 - E_r) + 2\lambda \frac{\kappa_P}{\chi_R} \mathbf{u} \cdot \nabla E_r, \quad (33)$$

$$\frac{\partial E_r}{\partial t} = c\kappa_P(aT^4 - E_r^{(0)}) + \nabla \cdot \left( \frac{c\lambda}{\chi_R} \nabla E_r \right) \quad (34)$$

$$= c\kappa_P(aT^4 - E_r) - 2\lambda \frac{\kappa_P}{\chi_R} \mathbf{u} \cdot \nabla E_r + \nabla \cdot \left( \frac{c\lambda}{\chi_R} \nabla E_r \right), \quad (35)$$

where  $e$  is the specific internal energy. The term  $c\kappa_P(aT^4 - E_r^{(0)})$  represents the energy exchange in the comoving frame between the material and radiation through absorption and emission of radiation. The term  $2\lambda(\kappa_P/\chi_R)\mathbf{u} \cdot \nabla E_r$  is due to the Lorentz transformation of radiation energy density. We do not directly solve

Eqs. 32 & 34 because of our mixed-frame approach. Instead, we solve Eqs. 33 & 35. An implicit treatment is usually necessary in order to solve these equations because of their stiffness. However, the Lorentz transformation term  $2\lambda(\kappa_P/\chi_R)\mathbf{u} \cdot \nabla E_r$  can be treated explicitly in many situations because it is of similar order to the term  $\lambda\mathbf{u} \cdot \nabla E_r$  in the hyperbolic subsystem (Eqs. 21 & 22), unless the Planck mean is much larger than the Rosseland mean. Without scattering, the Planck mean is usually larger than the Rosseland mean because the latter gives more weight to the lower opacity part of the radiation spectrum. However, in many astrophysical phenomena (e.g., shock breakout in core-collapse supernovae), electron scattering, which contributes to the Rosseland mean only, is the dominant source of opacity, and therefore the Lorentz transformation term can be neglected in those cases.

### 3. SINGLE-LEVEL INTEGRATION ALGORITHM

For each step at a single level of refinement, the state is first evolved using an explicit Godunov method for the hyperbolic subsystem (§ 3.1). Then an implicit update for radiation diffusion and source-sink terms is performed (§ 3.2).

It is customary in time-split schemes to denote intermediate quantities with a fractional time index such as  $n + 1/2$ , so that, for example, the explicit hyperbolic update would advance radiation energy density from  $E_r^n$  to  $E_r^{n+1/2}$  and the implicit update would then advance it from  $E_r^{n+1/2}$  to  $E_r^{n+1}$ . We are not using this notational convention here mainly to avoid confusion in the following section with time-centered quantities constructed at the actual intermediate time  $t^{n+1/2}$ . In § 3.2, where we write out the implicit update in detail, we will refer to the post-hyperbolic intermediate quantities as  $E_r^-$ ,  $(\rho e)^-$ , etc.

#### 3.1. Explicit Solver for Hyperbolic subsystem

The hyperbolic subsystem is treated explicitly. This explicit part of our numerical integration algorithm for radiation hydrodynamics is very similar to the hydrodynamics algorithm presented in Paper I of this series. We refer the reader to Paper I for detailed description of the integration scheme, which supports a general equation of state, self-gravity, and nuclear reactions. Here we will only present the parts specific to radiation hydrodynamics.

The advection part of the time evolution can be written in the form

$$\frac{\partial \mathbf{U}}{\partial t} = -\nabla \cdot \mathbf{F}, \quad (36)$$

where  $\mathbf{U} = (\rho, \rho\mathbf{u}, \rho E, E_r)^T$  with the superscript  $T$  denoting the transpose operation are the conserved variables, and  $\mathbf{F}$  is their flux. The conserved variables are defined at cell centers. We predict the primitive variables, including  $\rho$ ,  $\mathbf{u}$ ,  $p$ ,  $\rho e$ ,  $E_r$ , from cell centers at time  $t^n$  to edges at time  $t^{n+1/2}$  and use an approximate Riemann solver to construct fluxes,  $\mathbf{F}^{n+1/2}$ , on cell faces. This algorithm is formally second-order in both space and time. The time step is computed using the standard CFL condition for explicit methods, with additional

constraints if additional physics (such as burning) is included. The sound speed used in the computation is now the radiation modified sound speed  $c_s$  (Eq. 28).

#### 3.1.1. Construction of Fluxes

CASTRO solves the hyperbolic subsystem of radiation hydrodynamics with an unsplit piecewise parabolic method (PPM) with characteristic tracing and full corner coupling (Miller & Colella 2002). The four major steps in the construction of the face-centered fluxes,  $\mathbf{F}^{n+1/2}$ , in the case of hydrodynamics have been described in details in Paper I. The extension to radiation hydrodynamics is straightforward given the characteristic analysis presented in § 2.2. Thus, we will not repeat the details here. We will also omit the procedures for passively advected quantities, auxiliary variables, gravity, and reaction, because they do not change.

First, we compute the primitive variables defined as  $\mathbf{Q} \equiv (\rho, \mathbf{u}, p, \rho e, E_r, p_{\text{tot}}, \rho e + E_r)^T$ . Note that several variables are redundant for various reasons (i.e., efficiency and safety), which will be explained later.

Second, we reconstruct parabolic profiles of the primitive variables within each cell. The total pressure,  $p_{\text{tot}}$ , rather than the gas pressure,  $p$ , is used in computing a flattening coefficient (Miller & Colella 2002).

In the third step, we perform characteristic extrapolations of the primitive variables and obtain the edge values of  $\mathbf{Q}$  at  $t^{n+1/2}$  using the eigenvectors of the system (Eq. 29 & 30) and the parabolic profiles of the primitive variables. Flattening is applied in this procedure.

Finally, the fluxes are computed for the edge values obtained in the last step using an approximate Riemann solver, which is based on the Riemann solver of Bell et al. (1989) and Colella et al. (1997). The computational procedure is essentially the same as that in Paper I except that the gas internal energy density,  $\rho e$ , and gas pressure,  $p$  are now replaced by the total internal energy density,  $\rho e + E_r$ , and total pressure,  $p + \lambda E_r$ . The Riemann solver computes the Godunov state at the interface, which is then used to compute the fluxes,  $(\rho\mathbf{u}, \rho\mathbf{u}\mathbf{u} + p\mathbf{I}, \rho E\mathbf{u} + p\mathbf{u}, ((3-f)/2)E_r\mathbf{u})^T$ . Recall that there are redundant variables in the primitive variables,  $\mathbf{Q}$ , for efficiency and safety. With the total internal energy density,  $\rho e + E_r$ , a call to the equation of state can be avoided. Negative radiation energy density and gas pressure can also be avoided in the Godunov state. The term,  $\lambda\mathbf{u} \cdot \nabla E_r$ , in Eqs. 21 & 22 is computed as follows,

$$\begin{aligned} (\lambda\mathbf{u} \cdot \nabla E_r)_{i,j,k} &= \lambda_{i,j,k} \\ &\times \left[ \left( \frac{u_{x,i-1/2,j,k} + u_{x,i+1/2,j,k}}{2} \right) \left( \frac{E_{r,i+1/2,j,k} - E_{r,i-1/2,j,k}}{\Delta x} \right) \right. \\ &+ \left( \frac{u_{y,i,j-1/2,k} + u_{y,i,j+1/2,k}}{2} \right) \left( \frac{E_{r,i,j+1/2,k} - E_{r,i,j-1/2,k}}{\Delta y} \right) \\ &\left. + \left( \frac{u_{z,i,j,k-1/2} + u_{z,i,j,k+1/2}}{2} \right) \left( \frac{E_{r,i,j,k+1/2} - E_{r,i,j,k-1/2}}{\Delta z} \right) \right], \end{aligned} \quad (37)$$

where the variables with half-integer index are the Godunov states. The term,  $\lambda\nabla E_r$ , in Eq. 20 is computed in a similar way.

Depending upon a switch set by the user, the Lorentz transformation term,  $2\lambda(\kappa_P/\chi_R)\mathbf{u} \cdot \nabla E_r$ , may be included in the explicit update.



### 3.2. Implicit Solver for Radiation Diffusion and Source-Sink Terms

The implicit solver evolves the radiation and gas according to Eqs. 33 & 35. The algorithm uses a first-order backward Euler discretization. We note that the Lorentz term,  $2\lambda(\kappa_P/\chi_R)\mathbf{u} \cdot \nabla E_r$ , may or may not be included in the implicit solver. The advantage of including this term here is that it effectively balances emission against absorption in the comoving frame as the material becomes optically-thick and this coupling becomes stiff. Implicit treatment also helps avoid a time step restriction when  $\kappa_P/\chi_R$  is significantly greater than 1. The disadvantage is that it makes the resulting linear systems nonsymmetric, but this not a major concern in practice.

The radiation update algorithm is based on that of Howell & Greenough (2003). The update from the post-hyperbolic state to time  $t^{n+1}$  for Eqs. 33 & 35 has the form

$$\frac{\rho e^{n+1} - \rho e^-}{\Delta t} = -c\kappa_P^{n+1} [a(T^{n+1})^4 - E_r^{n+1}] + q^{n+1} \mathbf{u} \cdot \nabla E_r^{n+1}, \quad (38)$$

$$\frac{E_r^{n+1} - E_r^-}{\Delta t} = +c\kappa_P^{n+1} [a(T^{n+1})^4 - E_r^{n+1}] - q^{n+1} \mathbf{u} \cdot \nabla E_r^{n+1} + \nabla \cdot (d^{n+1} \nabla E_r^{n+1}), \quad (39)$$

where  $q^{n+1} = 2\lambda^{n+1}\kappa_P^{n+1}/\chi_R^{n+1}$  and  $d^{n+1} = c\lambda^{n+1}/\chi_R^{n+1}$ . Here, we use the  $-$  superscript to denote the state following the explicit update, and the  $n+1$  superscript for the state at  $t^{n+1}$ . Since the velocity does not change in the implicit radiation update, we have dropped the  $-$  superscript for  $\mathbf{u}$ . We solve Eqs. 38 & 39 iteratively via Newton's method. We define

$$F_e = \rho e^{n+1} - \rho e^- - \Delta t \{ -c\kappa_P^{n+1} [a(T^{n+1})^4 - E_r^{n+1}] + q^{n+1} \mathbf{u} \cdot \nabla E_r^{n+1} \}, \quad (40)$$

$$F_r = E_r^{n+1} - E_r^- - \Delta t \{ c\kappa_P^{n+1} [a(T^{n+1})^4 - E_r^{n+1}] - q^{n+1} \mathbf{u} \cdot \nabla E_r^{n+1} + \nabla \cdot (d^{n+1} \nabla E_r^{n+1}) \}. \quad (41)$$

Here, we have dropped the  $-$  superscript for  $\rho^-$  because the implicit update does not change the mass density. The desired solution is for  $F_e$  and  $F_r$  to both be zero, and the Newton update to approach this state is the solution to the linear system

$$\begin{bmatrix} (\partial F_e / \partial T)^{(k)} & (\partial F_e / \partial E_r)^{(k)} \\ (\partial F_r / \partial T)^{(k)} & (\partial F_r / \partial E_r)^{(k)} \end{bmatrix} \begin{bmatrix} \delta T^{(k+1)} \\ \delta E_r^{(k+1)} \end{bmatrix} = \begin{bmatrix} -F_e^{(k)} \\ -F_r^{(k)} \end{bmatrix}. \quad (42)$$

Here  $\delta T^{(k+1)} = T^{n+1,(k+1)} - T^{n+1,(k)}$  and  $\delta E_r^{(k+1)} = E_r^{n+1,(k+1)} - E_r^{n+1,(k)}$ , where the  $(k)$  superscript denotes the stage of the Newton iteration. To reduce clutter we drop the  $n+1$  superscript without loss of clarity, so  $E_r^{(k+1)} \equiv E_r^{n+1,(k+1)}$ .

To solve this system we eliminate the dependency on  $\delta T$  by forming the Schur complement, leaving a modified diffusion equation for the radiation update  $\delta E_r$ . For simplicity we drop the temperature derivatives of  $d$  and  $q$ , keeping only the temperature dependence of the emission and absorption terms. This does not affect the converged solution and in practice does not appear to significantly

degrade the convergence rate.

After some mathematical manipulation we obtain the following diffusion equation, which must be solved for each Newton iteration:

$$\begin{aligned} & \left[ (1 - \eta)c\kappa_P + \frac{1}{\Delta t} \right] E_r^{(k+1)} - \nabla \cdot (d \nabla E_r^{(k+1)}) \\ & + (1 - \eta)q\mathbf{u} \cdot \nabla E_r^{(k+1)} \\ & = (1 - \eta)c\kappa_P a (T^{(k)})^4 + \frac{1}{\Delta t} [E_r^- - \eta(\rho e^{(k)} - \rho e^-)], \end{aligned} \quad (43)$$

where

$$\eta = 1 - \frac{\rho c_v}{\rho c_v + c\Delta t \frac{\partial}{\partial T} [\kappa_P (aT^4 - E_r)]} \approx -\frac{\partial F_r}{\partial T} \left( \frac{\partial F_e}{\partial T} \right)^{-1}, \quad (44)$$

and  $c_v$  is the specific heat capacity of the matter.

The iterations are stopped when the maximum of  $|\delta E_r^{(k+1)} / E_r^{n+1,(k+1)}|$  on the computational domain falls below a preset tolerance (e.g.,  $10^{-6}$ ). Note that Eq. 43 is rewritten to be in terms of  $E_r^{n+1,(k+1)}$  rather than  $\delta E_r^{(k+1)}$ . This is done for computational efficiency. After one or two Newton iterations, the solution at the previous iteration,  $E_r^{n+1,(k)}$ , is getting very close to the final converged solution  $E_r^{n+1}$ . Since we use  $E_r^{n+1,(k)}$  as the starting point for the call to the iterative linear solver these calls get cheaper for each additional Newton iteration. If we solved for  $\delta E_r^{(k+1)}$  instead, we would have to change the linear solver tolerance to avoid an unnecessarily accurate and expensive solve for what may be a very small correction.

Each time we solve the diffusion equation for a new iterate  $E_r^{n+1,(k+1)}$ , we update the gas internal energy density as follows,

$$\begin{aligned} \rho e^{(k+1)} &= \eta \rho e^{(k)} + (1 - \eta) \rho e^- \\ &- \Delta t (1 - \eta) [c\kappa_P (a(T^{(k)})^4 - E_r^{(k+1)}) - q\mathbf{u} \cdot \nabla E_r^{(k+1)}]. \end{aligned} \quad (45)$$

The new temperature  $T^{n+1,(k+1)}$  then derives from  $e^{n+1,(k+1)}$  and  $\rho$  via a call to the equation of state. Other quantities may or may not be updated: the coefficients  $\kappa_P$ ,  $\chi_R$ ,  $\eta$ ,  $d$ , and  $q$  are also temperature-dependent and could have been written with a  $(k)$  superscript. The limiter  $\lambda$  can be recomputed based on the new  $E_r^{n+1,(k+1)}$ . It is a tradeoff between efficiency and accuracy whether to recompute some or all of these at every iteration. Our default choice is to recompute all of these at each iteration for better accuracy. However, updating coefficients can make the implicit update iteration less stable if the coefficients are not smooth functions of their inputs, in addition to the extra computational costs. But since each stage of the Newton iteration (Eq. 43 combined with Eq. 45) is itself a conservative solution, the implicit algorithm will conserve total energy to the tolerance of the linear solver (which should not be confused with the tolerance of the Newton iterations) regardless of the number of iterations taken or which coefficients are updated.

In CASTRO, the linear system Eq. 43 is solved using the *hypr* library (Falgout & Yang 2002;

*hypr* Code Project 2011). We have developed drivers that work with systems in the canonical form

$$AE_r^{n+1} - \sum_i \frac{\partial}{\partial x^i} \left( B_i \frac{\partial E_r^{n+1}}{\partial x^i} \right) + \sum_i \frac{\partial}{\partial x^i} (C_i E_r^{n+1}) + \sum_i D_i \frac{\partial E_r^{n+1}}{\partial x^i} = \text{rhs}, \quad (46)$$

where  $A$  are cell-centered coefficients and  $B_i$ ,  $C_i$ , and  $D_i$  are centered at cell faces. For Eq. 43 the  $C_i$  coefficients are not used. There is some subtlety in the appropriate averaging of coefficients from cells to faces; see Howell & Greenough (2003) for further discussion. The same canonical form works for Cartesian, cylindrical, and spherical coordinates so long as appropriate metric factors are included in the coefficients and the rhs.

With *hypr* we have a choice between two parallel multigrid solvers: Schaffer multigrid (SMG) and PFMG. SMG is more robust for difficult problems with strongly-varying coefficients, but PFMG is typically more efficient and scalable. These solvers work for systems at a uniform grid resolution (that is, systems associated with a single level of adaptive mesh refinement). For systems coupling together more than one refinement level we could use the *hypr* algebraic multigrid (AMG) solver, or an FAC-type scheme (McCormick 1989) using structured solvers on the separate levels. In earlier versions of the AMR algorithm we required multilevel solvers for conservative coupling between refinement levels. We have now developed a scheme, though, that eliminates the need for a multilevel linear solver while still conserving total energy. We discuss this in detail in the following sections. Note also that use of the  $D_i$  coefficients deriving from the Lorentz term make the diffusion equation nonsymmetric. The multigrid solvers mentioned above are designed for use with symmetric systems, but good convergence behavior can still be obtained by using these solvers as preconditioners for a Krylov method such as GMRES.

#### 4. AMR

CASTRO uses a nested hierarchy of logically-rectangular, variable-sized grids with simultaneous refinement in both space and time, as illustrated in Figures 1 and 2. One major design objective of the AMR algorithm is to preserve the conservation properties of the uniform-grid discretization. AMR for hyperbolic equations in CASTRO was described in detail in Paper I.

The explicit update step for radiation hydrodynamics (§ 3.1) follows the same pattern as other hyperbolic equations and so does not increase the complexity of the AMR algorithm. We note that the hyperbolic subsystem (Eqs. 19 – 22) is only partially in conservation law form. It will not conserve total momentum because it does not include an equation analogous to Eq. 5 tracking the radiation momentum. It will conserve total energy, though, so long as the divergence terms are differenced in a conservative manner. These divergence terms therefore require AMR reflux operations, as described in Berger & Colella (1989). The term  $\lambda \mathbf{u} \cdot \nabla E_r$  appears in both the gas energy and radiation energy equations with opposite signs, so any consistent discretization of these terms will conserve total energy.

The AMR version of the implicit radiation diffusion up-

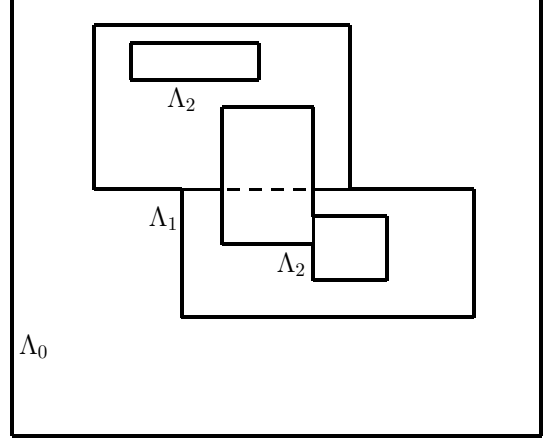


FIG. 1.— A properly nested hierarchy of grids. Each grid consists of a number of cells. Thick lines represent level boundaries. The union of fine grids at level  $\ell$  is contained within the union of coarser grids at level  $\ell - 1$ .

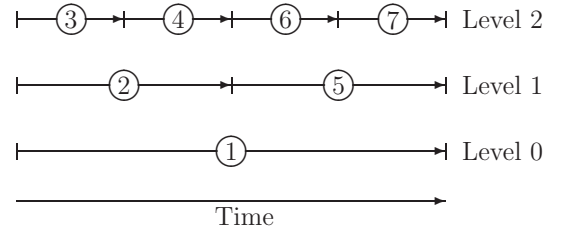


FIG. 2.— One coarse time step for an adaptive run with one base level and two refinement levels. The numbers mark the order of the steps.

date is based on Howell & Greenough (2003), but with several important differences: The present algorithm is fully-implicit, not time-centered. The optional multilevel linear solve at the beginning of each coarse time step is no longer included—this feature was introduced to improve accuracy but we now consider it unnecessary in most cases. Finally, the multilevel linear solve for flux synchronization between coarse and fine levels is replaced by a new algorithm we call the “deferred sync.” These changes entirely eliminate the need to compute linear system solutions coupling different levels of the AMR hierarchy, while not compromising conservation of total energy. Performance is significantly improved because multilevel linear solvers tend to be more complex and expensive than those for single-level systems.

##### 4.1. AMR Time Step Outline with Deferred Sync

The AMR time step is defined recursively in terms of operations on a level  $\ell$  and its interactions with coarser and finer levels. We consider advancing level  $\ell$  from time index  $n$  to  $n + 1$ , corresponding to time values  $t^{\text{old},\ell}$  and  $t^{\text{new},\ell}$ , respectively. (Even though levels other than  $\ell$  have executed different numbers of time steps, we will use the  $n + 1$  superscript to refer to values at time  $t^{\text{new},\ell}$  on all levels involved in the calculation.) The region covered by level  $\ell$  is denoted  $\Lambda^\ell$ , its border is  $\partial\Lambda^\ell$ , and the border of the next finer level, projected onto level  $\ell$ , is  $\mathbf{P}(\partial\Lambda^{\ell+1})$ .

The notation to describe all of this is unavoidably complex due to the quantities at different times, levels, and stages of the update process. In the following outline, we

specify the update for the level  $\ell$  and its synchronization with finer levels. We include the hyperbolic update and the refluxing step associated with it in order to show the proper sequence of operations and to contrast the explicit reflux with the implicit deferred sync. As in § 3.1 the hyperbolic conserved state vector is denoted by  $\mathbf{U}$ , but we denote the hyperbolic flux by  $\mathbf{F}_H$  to distinguish it from the radiation flux. For the radiation flux we are concerned here only with the diffusion term in the implicit update, and we denote the associated flux by  $\mathbf{F}_R$  to distinguish it from the complete radiation flux  $\mathbf{F}_r$  introduced in § 2.1.

Note that while the flux divergence is needed everywhere, the fluxes  $\mathbf{F}_H$  and  $\mathbf{F}_R$  themselves are stored only on the borders between levels. Our code has data structures called flux registers designed for this purpose. The notation  $\langle \cdot \rangle$  indicates an average of level  $\ell + 1$  data in space over the fine cells (or cell faces) making up each corresponding coarse cell (or face), while  $\langle \langle \cdot \rangle \rangle$  denotes an average of level  $\ell + 1$  data in both space and time over the coarse (level  $\ell$ ) time step. Thus, at the only point in the algorithm where this notation is used,

$$\langle \langle \mathbf{F}^{\ell+1} \rangle \rangle = \frac{1}{r^{\ell+1}} \sum_{m=r^{\ell+1}n}^{r^{\ell+1}(n+1)-1} \langle \mathbf{F}^{\ell+1,m+1} \rangle, \quad (47)$$

where  $r^{\ell+1}$  is the number of level  $\ell + 1$  time steps making up the level  $\ell$  time step from  $n$  to  $n + 1$ , and  $\mathbf{F}^{\ell+1,m+1}$  is the level  $\ell + 1$  flux during the level  $\ell + 1$  time step  $m + 1$ .

The expression involving  $\nabla_{\text{Reflux}} \cdot$  that appears as a deferred source term in the diffusion equation and explicitly for the hyperbolic reflux is called the reflux divergence, because it takes the form of the divergence of a flux difference  $\delta \mathbf{F}$  stored in the flux registers. These terms are evaluated *only* in the coarse cells bordering an interface with a finer level. They do not affect fine cells at the interface because flux calculations for those cells are already the more accurate ones; the  $\nabla_{\text{Reflux}} \cdot$  terms represent the corrected fluxes from these fine cells being imposed onto the coarse grid.

Another way to understand the  $\nabla_{\text{Reflux}} \cdot$  terms is to consider  $\delta \mathbf{F}_H$  and  $\delta \mathbf{F}_R$  to be energy that has been “misplaced” at the coarse-fine interfaces during the level time step, due to the differing flux calculations on the different levels. If the solution were not corrected, this energy would be lost and the system would not be conservative. Instead, the  $\nabla_{\text{Reflux}} \cdot$  terms re-introduce the missing energy into the system. For the explicit hyperbolic flux this is done explicitly; for the radiation flux it contributes to the right hand side for the implicit update, so as not to impose a new stability constraint on the size of the time step.

The hyperbolic state vector  $\mathbf{U}$  includes the radiation and fluid energies updated in the implicit update step, but there is no ambiguity because the operations and their associated fluxes are completely distinct. The refluxing update to  $\mathbf{U}$  is written as an update, with  $\mathbf{U}$  appearing on both sides of the equation, because otherwise we would need additional notation to indicate the pre- and post-reflux states. Averaging down from fine to coarse levels is also written as an update. The meaning of the rest of the pseudocode below should be reasonably clear in context:

**If** ( $\ell < \ell_{\max}$ ) and (regrid requested from base level  $\ell$ )  
**then**

**For**  $\ell' \in \{\ell_{\max} - 1, \dots, \ell\}$  **do**

- Determine new grid layout for level  $\ell' + 1$ .
- Interpolate data to new grids from level  $\ell'$ .
- Copy data on intersection with old level  $\ell' + 1$ .

**Enddo**

**Endif**

**Level Time Step**, level  $\ell$ :

**Explicit Hyperbolic Update** for  $\mathbf{U}^{\ell,n+1}$

(see Paper I)

- Set  $\mathbf{F}_H^{\ell,n+1}$  for hyperbolic fluxes  
on  $(\mathbf{P}(\partial\Lambda^{\ell+1}), \ell < \ell_{\max})$  and on  $(\partial\Lambda^{\ell}, \ell > 0)$

**Implicit Diffusion Update**

$$\begin{aligned} * \frac{E_r^{n+1} - E_r^-}{\Delta t^{n+1}} &= +c\kappa_P^{n+1} [a(T^{n+1})^4 - E_r^{n+1}] \\ &\quad - q^{n+1} \mathbf{u} \cdot \nabla E_r^{n+1} \\ &\quad + \nabla \cdot (d^{n+1} \nabla E_r^{n+1}) \\ &\quad - (\Delta t^n / \Delta t^{n+1}) \nabla_{\text{Reflux}} \cdot (\delta \mathbf{F}_R^{\ell+1,n}) \quad \text{on } \Lambda^{\ell} \\ * \frac{\rho e^{n+1} - \rho e^-}{\Delta t^{n+1}} &= -c\kappa_P^{n+1} [a(T^{n+1})^4 - E_r^{n+1}] \\ &\quad + q^{n+1} \mathbf{u} \cdot \nabla E_r^{n+1} \quad \text{on } \Lambda^{\ell} \end{aligned}$$

**End Implicit Diffusion Update**

- $\mathbf{F}_R^{\ell,n+1} = -d^{n+1} \nabla E_r^{n+1}$   
on  $(\mathbf{P}(\partial\Lambda^{\ell+1}), \ell < \ell_{\max})$  and on  $(\partial\Lambda^{\ell}, \ell > 0)$
- Advance levels  $\ell + 1, \dots, \ell_{\max}$  recursively.
- $\delta \mathbf{F}_H^{\ell+1,n+1} = \langle \langle \mathbf{F}_H^{\ell+1} \rangle \rangle - \mathbf{F}_H^{\ell,n+1}$   
on  $\mathbf{P}(\partial\Lambda^{\ell+1}), \ell < \ell_{\max}$
- $\delta \mathbf{F}_R^{\ell+1,n+1} = \langle \langle \mathbf{F}_R^{\ell+1} \rangle \rangle - \mathbf{F}_R^{\ell,n+1}$   
on  $\mathbf{P}(\partial\Lambda^{\ell+1}), \ell < \ell_{\max}$

**End Level Time Step**

**If** ( $\ell < \ell_{\max}$ ) **then**

(synchronization/refluxing between levels  $\ell$  and  $\ell + 1$ )

- $\mathbf{U}^{\ell,n+1} := \mathbf{U}^{\ell,n+1} - (\Delta t^{n+1}) \nabla_{\text{Reflux}} \cdot (\delta \mathbf{F}_H^{\ell+1,n+1})$   
on  $\Lambda^{\ell} - \mathbf{P}(\Lambda^{\ell+1})$
- $\mathbf{U}^{\ell,n+1} := \langle \mathbf{U}^{\ell+1,n+1} \rangle$  on  $\mathbf{P}(\Lambda^{\ell+1})$

**Endif** (end synchronization/refluxing).

The advantages of the deferred sync algorithm are that it eliminates the need for a separate linear solve for synchronization and eliminates the need for solving multi-level linear systems entirely. It does, however, add complications of its own to the AMR implementation. One is that there is no longer any point within the time step cycle when the field variables are fully synchronized. At the end of a coarse time step all levels have reached the same point in time, but if we want to actually compute the energy budget and confirm conservation we have to include the contributions from deferred fluxes stored in flux registers. These will not be re-introduced into the state until the next time step. The end of a coarse time step is also the natural time for checkpoint/restart operations, so to reproduce the saved state of the system we now have to include the deferred fluxes in checkpoints as

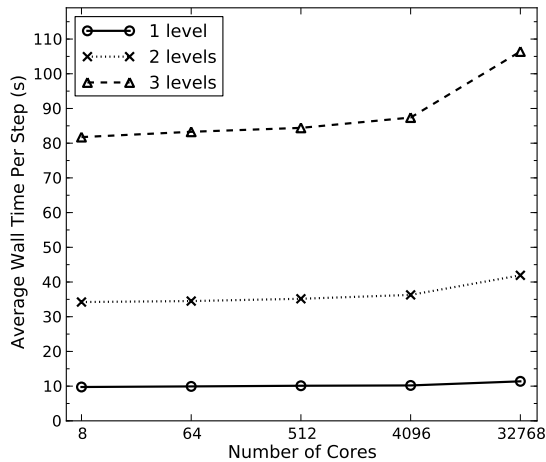


FIG. 3.— Weak scaling behavior of CASTRO on Hopper at NERSC. Average wall clock times per coarse time step are shown for simulations with 1 (*circle*), 2 (*X symbol*), and 3 total grid levels (*triangle*). The numbers of cells that are advanced in one coarse time step increase by a factor of three and seven, for the two- and three-level simulations, respectively.

well.

Regridding becomes an issue as well. The adaptive algorithm periodically re-evaluates the refinement criteria for each level and may change the layout of refined grids. For grids at level  $\ell$  the criteria are evaluated at level  $\ell - 1$ , and this happens between level  $\ell - 1$  time steps. The interpolation operations between coarse and fine field variables are conservative. With the deferred sync, though, there will also be fluxes stored around the edges of level  $\ell$  at the time that level may be changed. There is no straightforward way to transform these stored fluxes so that they coincide with the new mesh layout. Instead, what we have is an old set of flux registers that may now overlap with level  $\ell$  as well as with level  $\ell - 1$ , and if the grid layout changes enough may even overlap with other levels both finer and coarser. The deferred sync idea still applies, but the implementation becomes more complicated than the pseudocode above suggests. Portions of the stored flux at the interface between levels  $\ell$  and  $\ell - 1$  may be reintroduced into the level  $\ell$  advance or the level  $\ell + 1$  advance, and so on, not just into level  $\ell - 1$ .

## 5. PARALLEL PERFORMANCE

CASTRO is implemented within the BoxLib framework for parallel structured-grid AMR applications (Paper I and references therein). In BoxLib, parallelization is based upon either hybrid OpenMP-MPI or pure MPI. Because the *hyple* library does not fully support OpenMP yet, we use the pure MPI approach for the radiation hydrodynamics solver. For more information on software design and parallelization, we refer the reader to Paper I. Here we show the scaling behavior of the radiation hydrodynamics solver in CASTRO.

A weak scaling study has been carried out on Hopper<sup>4</sup>, a petascale Cray XE6 supercomputer at the National Energy Research Scientific Computing Center. We have performed a series of three-dimensional simulations with 1, 2 and 3 total levels on various numbers of cores. For

the convenience of comparison, each run has one grid of  $64^3$  cells at each level on each core. A refinement factor of 2 is used in the multi-level simulations. Thus, the level 1 and 2 grids occupy 12.5% and 1.5625% of the whole volume, respectively. A point explosion like the one in § 6.9 is replicated on each core. The fine grids are placed at the center of the local domain of each core. Figure 3 shows that CASTRO has very good scaling behavior up to 32768 cores. For the single-level simulations, the average wall clock time per coarse time step increases by only 17% from 8 cores to 32768 cores. Because of subcycling in time for simulations with multiple levels, one coarse time step consists of one step on the coarse level and two steps on the next fine level, and the two fine steps might also consist of even finer steps, recursively. Thus, the number of cells that are advanced in one coarse time step increases by a factor of three or seven, for the two- and three-level simulations, respectively (Fig. 2). Our results also show that the overhead introduced by AMR is modest. For example, on 4096 cores, the average wall clock times per coarse time step for the two- and three-level simulations are 3.6 and 8.6 times more than that of the single-level simulation. This corresponds to an overhead of 19% and 22%, respectively. On 32768 cores, the average wall clock times per coarse time step for the two- and three-level simulations are 3.8 and 9.5 times more than that of the single-level simulation. This corresponds to an overhead of 25% and 36%, respectively. It should be noted that single-level simulations with equivalent uniform grids would cost  $\sim 4.5$  and 30 times more time than the corresponding two- and three-level simulations we have run even if the single-level simulations are assumed to scale perfectly. We also note that in this series of simulations about half of the time is spent on the implicit evolution of the parabolic part of the system.

## 6. TEST PROBLEMS

In this section we present detailed tests of the code demonstrating its ability to handle a wide range of radiation hydrodynamics problems. Note that not every term is included in every test so that the algorithms for these terms can be tested separately. We test the radiation source-sink term in isolation in the approach to thermal equilibrium test (§ 6.1). For a nonequilibrium Marshak wave problem (§ 6.2), the simulation involves the parabolic subsystem only. In § 6.3, we assess the accuracy of the AMR algorithms for radiation diffusion using a thermal wave test, and perform a convergence study. The system in a radiation front test (§ 6.4) is in the optically-thin streaming limit, whereas the system in a shock tube problem (§ 6.5) is in the limit of strong equilibrium with almost no diffusion. The full radiation hydrodynamics system is included in a nonequilibrium radiative shock problem (§ 6.6) and the advection of a radiation pulse problem (§ 6.7). In § 6.8, we demonstrate the ability of CASTRO to maintain a static equilibrium of the gas and radiation pressures. In a radiative blast wave test (§ 6.9), we compare the results of simulations in 1D spherical, 2D cylindrical ( $r$  and  $z$ ), and 3D Cartesian coordinates. Finally, we demonstrate the ability of CASTRO to handle a large Lorentz transformation term in another radiative blast wave test (§ 6.10).

A CFL number of 0.8 is used for these tests unless stated otherwise or a fixed time step is used. The refine-

<sup>4</sup> The Hopper supercomputer was named after Grace Hopper, a pioneer in computer science and the developer of the first compiler.



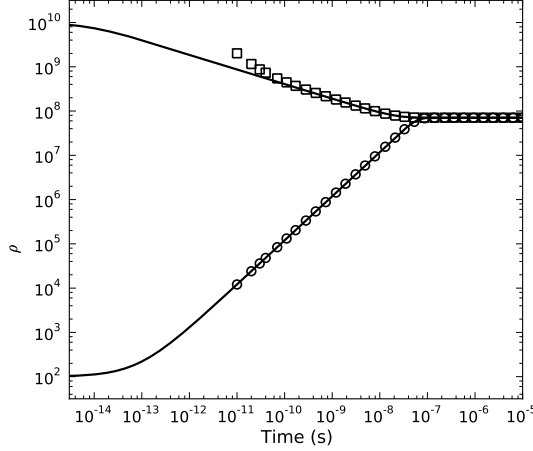


FIG. 4.— Evolution of internal energy density of gas for calculations of the approach to thermal equilibrium. Numerical results are shown in symbols, whereas the analytic solutions are shown in solid lines. Two cases with an initial internal energy density of  $10^{10} \text{ erg cm}^{-3}$  (upper solid line and squares) and  $10^2 \text{ erg cm}^{-3}$  (lower solid line and circles) are shown. The time step is fixed at  $\Delta t = 10^{-11} \text{ s}$ .

ment factor is 2 for all AMR runs. The relative tolerance for the Newton iterations in the implicit update is  $10^{-6}$  for all runs. The Lorentz transformation term is handled explicitly except in the second radiative blast wave test (§ 6.10).

### 6.1. Approach to Thermal Equilibrium

This test introduced by Turner & Stone (2001) has often been used to test the ability of a code to handle the source-sink term,  $c\kappa_P(aT^4 - E_r)$ . The test consists of a static uniform field of gas and radiation. The gas has a density of  $\rho = 10^{-7} \text{ g cm}^{-3}$ , a Planck mean absorption coefficient of  $\kappa_P = 4 \times 10^{-8} \text{ cm}^{-1}$ , a mean molecular weight of  $\mu = 0.6$ , and an adiabatic index of  $\gamma = 5/3$ . The initial radiation energy density is  $E_r = 10^{12} \text{ erg cm}^{-3}$  corresponding to a temperature of  $\sim 3.39 \times 10^6 \text{ K}$ . Two cases with an initial internal energy density of  $10^2 \text{ erg cm}^{-3}$  and  $10^{10} \text{ erg cm}^{-3}$ , respectively, have been studied. The gas temperatures are  $\sim 4.81 \text{ K}$  and  $\sim 4.81 \times 10^8 \text{ K}$  for the two cases, respectively. The time step is chosen as  $\Delta t = 10^{-11} \text{ s}$ . The evolution of the system will bring the gas and radiation into a thermal equilibrium. The radiation energy density will hardly change because the energy exchange between the gas and radiation is only a small fraction of the radiation energy. Therefore an analytic solution can be calculated by solving the following ordinary differential equation

$$\frac{d(\rho e)}{dt} = -c\kappa_P(aT^4 - E_r), \quad (48)$$

where  $E_r$  is assumed to be constant. The results of this test of the approach to thermal equilibrium are shown in Figure 4. The agreement with the analytic solution is good, especially in the first case. For the second case, the cooling in the numerical calculation is initially slower than that in the analytic solution for the first few steps in agreement with the results of Turner & Stone (2001). The slow cooling is a result of the *backward* Euler method and a relatively large time step at the beginning of the decay of the gas temperature.

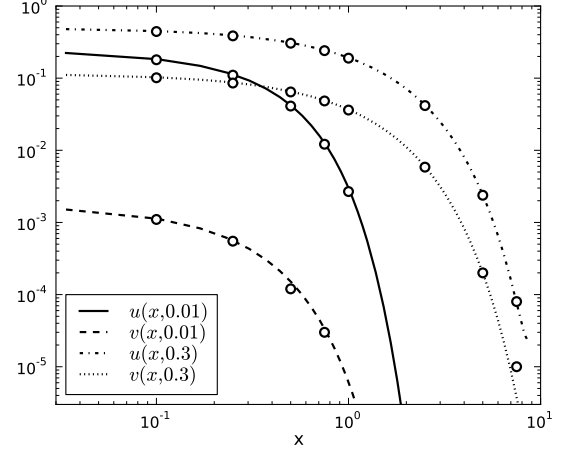


FIG. 5.— Nonequilibrium Marshak wave. Numerical results are shown in lines, whereas the analytic solutions are shown in circle symbols. We show the dimensionless radiation energy density  $u$  at  $\tau = 0.01$  (solid line) and  $\tau = 0.3$  (dash-dot line), and the dimensionless gas energy density  $v$  at  $\tau = 0.01$  (dashed line) and  $\tau = 0.3$  (dotted line). Here  $\tau$  is the dimensionless time.

### 6.2. Nonequilibrium Marshak Wave

In this test, we simulate the nonequilibrium Marshak wave problem in one dimension. Initially half of the space,  $z > 0$ , consists of a static uniform zero-temperature gas and no radiation. A constant radiation flux  $F_{\text{inc}}$  is incident on the surface at  $z = 0$ . The gas is not allowed to move in this idealized test. Thus the gas and radiation are coupled only through the source-sink term and the system is governed by Eqs. 33 & 35, with  $\mathbf{u} = 0$ . Su & Olson (1996) obtained analytic solutions for the problem under special assumptions. The matter is assumed to be gray with  $\kappa_P = \chi_R$ , and its volumetric heat capacity at constant volume is assumed to be  $c_v = \alpha T^3$ , where  $T$  is the gas temperature and  $\alpha$  is a parameter.

We have run this test with  $4a/\alpha = 0.1$  and no flux limiter (i.e.,  $\lambda = 1/3$ ). Figure 5 shows the numerical results of the dimensionless radiation energy density and gas energy density defined by Pomraning (1979)

$$x \equiv \sqrt{3}\kappa z, \quad (49)$$

$$\tau \equiv \left(\frac{4a c \kappa}{\alpha}\right) t, \quad (50)$$

$$u(x, \tau) \equiv \left(\frac{c}{4}\right) \left(\frac{E_r(z, t)}{F_{\text{inc}}}\right), \quad (51)$$

$$v(x, \tau) \equiv \left(\frac{c}{4}\right) \left(\frac{aT^4(z, t)}{F_{\text{inc}}}\right). \quad (52)$$

The computational domain of  $0 < x < 5\sqrt{3}$  is covered by 128 uniform cells. The dimensionless time step is chosen to be  $\Delta\tau = 3 \times 10^{-4}$ . The numerical results are in good agreement with the analytic results of Su & Olson (1996).

### 6.3. Thermal Wave

In this test, we simulate a thermal wave (Zel'Dovich & Raizer 1967) in three dimensions and use this test to assess the accuracy of the AMR algorithms for radiation diffusion. Suppose that a large amount of

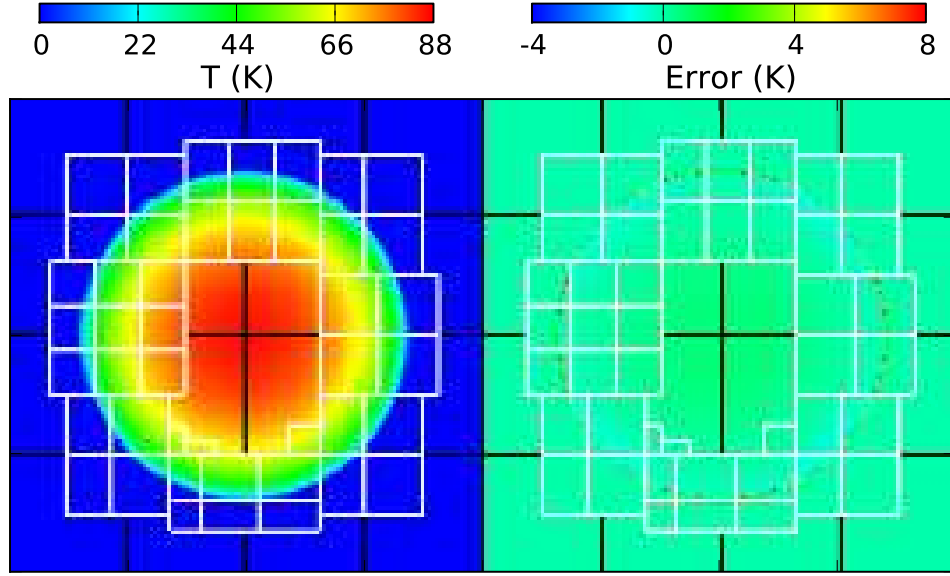


FIG. 6.— Temperature and error at  $t = 0.006$  s for the deferred sync run of the thermal wave test. The error is computed as the difference between the simulation results and the analytic solution. We show a 2D slice at  $x = 0$ . Also shown is the grid structure of the adaptive mesh. The physical domain in each dimension is  $(-200, 200)$  cm for each panel.

energy is deposited into a small volume as the internal energy of matter. The matter then starts to radiate and transfer most of its energy to radiation. We assume that the Planck mean absorption coefficient is large enough so that the matter and the radiation are in thermal equilibrium. The heat is transported out of the initial hot spot because of the nonlinear radiation heat conduction. As a result, a thermal wave develops. As the matter cools down, the matter gains most of the energy again. Initially the thermal wave speed is much higher than the sound speed. Assuming that there is no fluid motion and the matter contains most of the energy, there is an analytic solution for this problem (Zel'Dovich & Raizer 1967).

This test is adapted from Howell & Greenough (2003). The computational region in this test is three-dimensional with a domain of  $(-200, 200)$  cm in each dimension. Initially, the spherical hot spot has an energy of  $3 \times 10^7$  erg within a radius of 3.125 cm, whereas the ambient medium has a very low temperature of  $10^{-6}$  K. The volumetric heat capacity is  $\rho c_v = 0.05 \text{ erg K}^{-1} \text{ cm}^{-3}$ . The Planck mean absorption coefficient is  $\kappa_P = 10^6 \text{ cm}^{-1}$ , whereas the Rosseland mean is  $\chi_R = 10^{-3}(T/1 \text{ K})^{1/2} \text{ cm}^{-1}$ . In this test, the hydrodynamics is turned off, and there is no flux limiter (i.e.,  $\lambda = 1/3$ ). We have performed three simulations, a non-AMR run with  $128^3$  uniform cells, an AMR run using the deferred sync algorithm (§ 4.1), and an AMR run using the multilevel algorithm of Howell & Greenough (2003). The two AMR runs use 2 total levels with an effective resolution of  $128^3$  cells on the finer level. A cell is tagged for refinement if its temperature satisfies both  $\nabla T > 0.4T/\Delta x$  and  $T > 10^{-5}$  K, where  $\Delta x$  is the size of the cell. For the non-AMR run, we use a time step  $\Delta t = 1.03^{n-1} \times 5 \times 10^{-16}$  s for step  $n$ , whereas for the two AMR runs, we use  $\Delta t = 1.0609^{n-1} \times 1.015 \times 10^{-15}$  s for step  $n$  on the coarser level. We have chosen the time steps for the following reasons. First, as it expands, the

thermal wave is rapidly decelerated. Thus, a fixed time step would not be optimal. Instead, we let the time step grow over time. Second, the numerical error depends on both time step  $\Delta t$  and cell size  $\Delta x$ . In this test, we want to assess the accuracy of the AMR runs in comparison with a non-AMR run. Therefore, we make the time step on the finer level of the AMR runs to be roughly the same as that of the non-AMR run. Figure 6 shows a 2D slice at  $x = 0$  of the deferred sync run at  $t = 0.006$  s for temperature and the difference between the numerical and analytic results. It is shown that the largest errors ( $\sim 5$ – $10$  K) occur near the temperature profile is extremely steep. At the center, the absolute error is only 0.6 K, whereas the relative error is 0.7%. It takes 451 coarse time steps for the AMR runs to reach  $t = 0.006$  s. The finer grids in the AMR runs occupy 0.20, 2.3, and 32% of the volume at steps 1, 226, and 451, respectively. Thus, the benefit of AMR is obvious. To quantitatively measure the accuracy of the results, we compute the normalized  $L_1$ -norm error for temperature,  $\sum_{i,j,k} |T_{n,i,j,k} - T_a(t_n, x_i, y_j, z_k)| \Delta V_{i,j,k} / \sum_{i,j,k} T_a(t_n, x_i, y_j, z_k) \Delta V_{i,j,k}$ , where  $T_{n,i,j,k}$  and  $T_a(t_n, x_i, y_j, z_k)$  are the numerical and analytic results for cell  $(i, j, k)$ , respectively, and  $\Delta V_{i,j,k}$  is the volume of cell  $(i, j, k)$ . At  $t = 0.006$  s, the  $L_1$ -norm errors are 0.00792, 0.00833, and 0.00847, for the non-AMR, deferred sync, multilevel sync runs, respectively. The results show that our AMR algorithms have only increased the error by 5%–7%.

We have also performed a series of AMR simulations using the deferred sync to check the convergence behavior of the code. Besides the AMR run using the deferred sync that has been presented (Fig. 6), two lower resolution runs and one higher resolution run have been performed (Table 1). In all four runs, there are two total AMR levels. In the convergence study, when the cell size changes by a factor of 2 from one run to another, we

TABLE 1  
 $L_1$ -NORM ERRORS AND CONVERGENCE RATE  
 FOR THE THERMAL WAVE TEST PROBLEM.  
 FOUR 3D AMR RUNS WITH VARIOUS  
 RESOLUTIONS ARE SHOWN. THERE ARE TWO  
 AMR LEVELS IN EACH OF THESE RUNS.

Resolution <sup>a</sup>	$L_1$ Error	Convergence Rate
16, 32	0.0706	
32, 64	0.0245	1.5
64, 128	0.00833	1.6
128, 256	0.00268	1.6

<sup>a</sup> Number of cells across the width of the domain at each of the two AMR levels

change the time step by a factor of 4. Table 1 shows the  $L_1$ -norm errors and convergence rate at  $t = 0.006$  s. In this study, the order of accuracy with respect to  $\Delta x$  is  $\sim 1.6$ . Because our implicit scheme is first-order in time and second-order in the spatial discretization of the diffusion term, the expected convergence rate for a smooth flow is second-order with respect to  $\Delta x$  when  $\Delta t/\Delta x^2$  is fixed. However, the temperature profile of the thermal wave problem has a very steep slope near the front and its second derivative is discontinuous there. Hence, it is not surprising that the achieved order of accuracy is lower than 2.

#### 6.4. Optically-Thin Streaming of Radiation Front

In this problem, we test our code in the optically-thin streaming limit. This is the opposite limit from diffusion. The computational domain of this test is a one-dimensional region of  $0 < x < 100$  cm, covered by 128 uniform cells. Initially, the region  $x < 10$  cm is filled with radiation,  $E_r = 1 \text{ erg cm}^{-3}$ , whereas the region of  $x > 10$  cm has  $E_r = 10^{-10} \text{ erg cm}^{-3}$ . The left and right boundaries are held at  $E_r = 1$  and  $10^{-10} \text{ erg cm}^{-3}$ , respectively, during the calculation. The hyperbolic update (§ 3.1) is switched off in this test. The Planck mean absorption coefficient is set to zero. We have studied two cases with Rosseland means of  $\chi_R = 10^{-4}$  and  $10^{-7} \text{ cm}^{-1}$ , respectively. Thus the optical depths of the whole domain are  $10^{-2}$  and  $10^{-5}$ , respectively. For each case, we have performed two calculations, one with the time step set to  $\Delta t = \Delta x/(2c)$  and one with  $\Delta t = \Delta x/(20c)$ . Physically, the radiation front is expected to propagate at the speed of light, but the diffusion approximation does not naturally support a propagating front at any speed. Only the flux limiter permits the code to approximate the correct solution. In the numerical results the radiation front moves at approximately the speed of light with spreading due to diffusion (Figure 7). Better results are obtained for smaller time steps.

#### 6.5. Shock Tube Problem In Strong Equilibrium Regime

In this test, the one-dimensional numerical region ( $0 < x < 100$  cm) initially consists of two constant states:  $\rho_L = 10^{-5} \text{ g cm}^{-3}$ ,  $T_L = 1.5 \times 10^6 \text{ K}$ ,  $u_L = 0$  and  $\rho_R = 10^{-5} \text{ g cm}^{-3}$ ,  $T_R = 3 \times 10^5 \text{ K}$ ,  $u_R = 0$ , where  $L$  stands for the left state, and  $R$  the right state. The initial discontinuity is at  $x = 50$  cm. The gas is assumed to be ideal with an adiabatic index of  $\gamma = 5/3$  and a mean molecular weight of  $\mu = 1$ . Initially, the radiation is assumed to be in thermal equilibrium with the

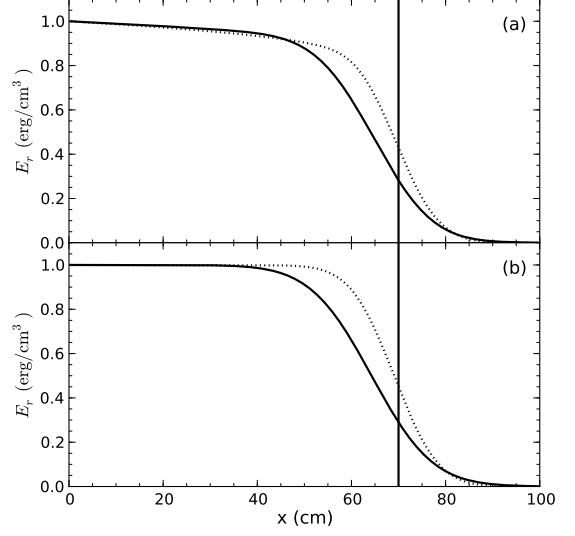


FIG. 7.— Optically-Thin Streaming of Radiation Front. We show radiation energy density at  $t = 2 \times 10^{-9}$  s for two cases: (a)  $\chi_R = 10^{-4} \text{ cm}^{-1}$  (upper panel); (b)  $\chi_R = 10^{-7} \text{ cm}^{-1}$  (lower panel). Results with a time step of  $\Delta t = \Delta x/(2c)$  are shown in solid lines, whereas those with a time step of  $\Delta t = \Delta x/(20c)$  in dotted lines. The vertical lines at  $x = 70$  cm indicate the expected position of the radiation front.

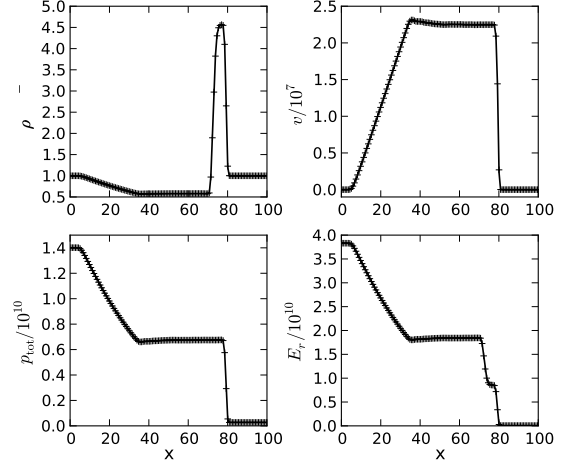


FIG. 8.— Shock tube at  $t = 10^{-6}$  s. Numerical results from a full radiation hydrodynamics calculation with 128 cells are shown in symbols. Results from a hydrodynamics calculation with 128 cells and an EOS for ideal gas plus radiation are shown in solid lines for comparison. We show mass density ( $\rho$ ), velocity ( $v$ ), total pressure ( $p_{\text{tot}}$ ), and radiation energy density ( $E_r$ ). The total pressure is the sum of gas pressure and radiation pressure. All quantities are in cgs units.

gas (i.e.,  $E_r = aT^4$ ). The interaction coefficients are set to  $\kappa_P = 10^6 \text{ cm}^{-1}$  and  $\chi_R = 10^8 \text{ cm}^{-1}$ . Thus, due to the huge opacities, the system is close to the limit of strong equilibrium with no diffusion (i.e.,  $E_r \approx aT^4$  and  $\chi_R \rightarrow \infty$ ), and is essentially governed by Eqs. 19–22. The parameters of this test problem are chosen such that neither radiation pressure nor gas pressure can be ignored. The numerical results are shown in Figure 8. Also shown are the results from a pure hydrodynamics calculation with an equation of state (EOS) for ideal gas plus radiation. As we expected, the results from the radiation hydrodynamics calculation are almost identical

to those of the pure hydrodynamics calculation. We also note that our scheme is stable without using small time steps even though the system is in the “dynamic diffusion” limit (Mihalas & Mihalas 1999; Krumholz et al. 2007) with  $\tau u/c \sim 10^7$ , where  $\tau$  is the optical depth of the system.

### 6.6. Nonequilibrium Radiative Shock

Radiation can modify the structure of a shock because it diffuses and because it interacts with matter. Analytic estimates of radiative shock structures can be found in Zel’Dovich & Raizer (1967); Mihalas & Mihalas (1999). Recently Lowrie & Edwards (2008) have found a semi-analytic exact solution of radiative shocks with gray nonequilibrium diffusion. In this section, we present our numerical results for two shock strengths and compare them to the solutions of Lowrie & Edwards (2008). The first problem is the Mach 2 shock in Lowrie & Edwards (2008). This is a subcritical shock in which the pre-shock matter is preheated by diffused radiation to a temperature that is lower than the temperature in the relaxation region far downstream. There is an embedded hydrodynamic shock that causes a jump in density and raises the temperature of the matter above the final downstream temperature. The matter temperature behind the embedded hydrodynamic shock then cools down in the relaxation region. The second problem is the Mach 5 supercritical shock in Lowrie & Edwards (2008). In the case of a supercritical shock, the matter temperature just before the hydrodynamic shock equals the downstream temperature. Most of the precursor region of a supercritical shock is close to equilibrium. Note that in both cases there is no jump in radiation energy density, otherwise the radiation flux would be infinite, which is unphysical.

The Mach 2 shock problem is simulated in a one-dimensional region of  $-1000 \text{ cm} < x < 500 \text{ cm}$  consisting of two constant states:  $\rho_L = 5.45887 \times 10^{-13} \text{ g cm}^{-3}$ ,  $T_L = 100 \text{ K}$ ,  $u_L = 2.35435 \times 10^5 \text{ cm s}^{-1}$  and  $\rho_R = 1.24794 \times 10^{-12} \text{ g cm}^{-3}$ ,  $T_R = 207.757 \text{ K}$ ,  $u_R = 1.02987 \times 10^5 \text{ cm s}^{-1}$ , where  $L$  stands for the left state, and  $R$  the right state. The initial discontinuity is at  $x = 0$ . The gas is assumed to be ideal with an adiabatic index of  $\gamma = 5/3$  and a mean molecular weight of  $\mu = 1$ . Initially, the radiation is assumed to be in thermal equilibrium with the gas (i.e.,  $E_r = aT^4$ ). The Planck and Rosseland coefficients are set to  $\kappa_P = 3.92664 \times 10^{-5} \text{ cm}^{-1}$  and  $\chi_R = 0.848902 \text{ cm}^{-1}$ , respectively. The boundaries are held at fixed values. Two refinement levels (three total levels) are used with the finest resolution at  $\Delta x \approx 2.9 \text{ cm}$ . In this test, a cell is tagged for refinement if the normalized second derivative of either density or temperature given by (Fryxell et al. 2000)

$$E_i = \frac{|u_{i+2} - 2u_i + u_{i-2}|}{|u_{i+2} - u_i| + |u_i - u_{i-2}| + \epsilon(|u_{i+2}| + 2|u_i| + |u_{i-2}|)} \quad (53)$$

is greater than 0.8. Here,  $\epsilon$  is a parameter set to 0.02 in this test, and  $u$  denotes either density or temperature. The initial conditions are set according to the pre-shock and post-shock states of the M2 shock. After a brief period of adjustment, the system settles down to a steady shock. The simulation is stopped at  $t = 0.05 \text{ s}$ . The results are shown in Figure 9. The agreement with the analytic solution is excellent, except that the numerical

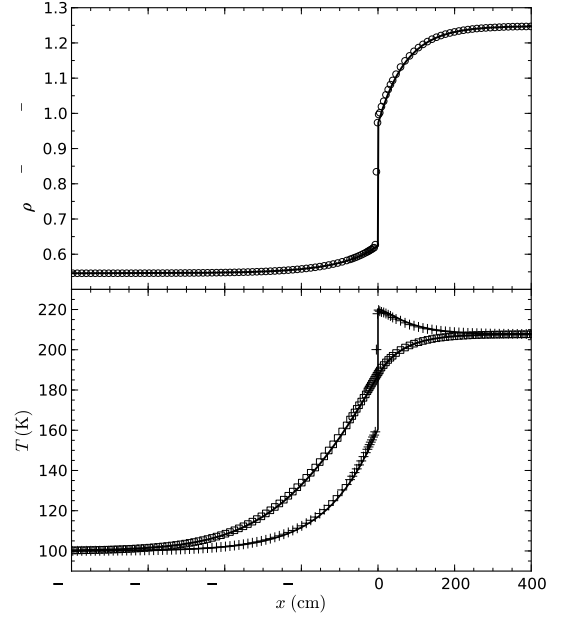


FIG. 9.— Density and temperature profiles for Mach 2 subcritical shock. Numerical results are shown in symbols. We show density (circle), gas temperature (plus sign), and radiation temperature (square). Here, radiation temperature is defined as  $(E_r/a)^{1/4}$ . The analytic solution is shown in solid lines. We show only part of the region near the hydro-shock. The numerical results have been shifted by +10 cm in space to compensate for the discrepancy in shock position caused by the initial setup.

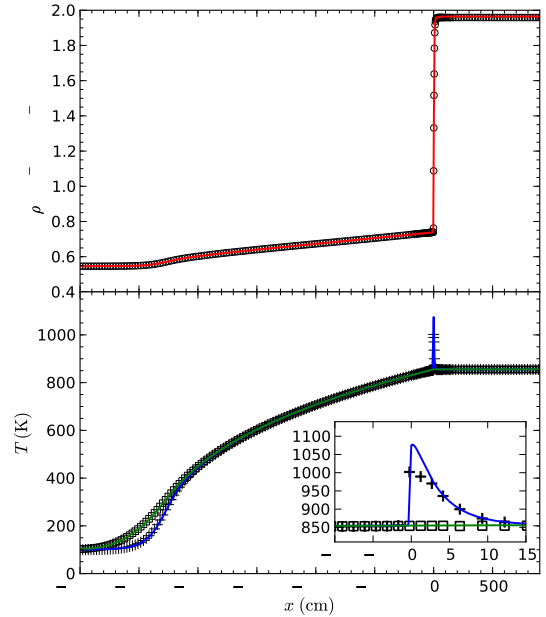


FIG. 10.— Density and temperature profiles for Mach 5 supercritical shock. Numerical results are shown in symbols. We show density (circle), gas temperature (plus sign), and radiation temperature (square). Here radiation temperature is defined as  $(E_r/a)^{1/4}$ . Also shown are the analytic results for density (red solid line), gas temperature (blue solid line), and radiation temperature (green solid line). We show only part of the region near the hydro-shock. The inset shows a blow-up of the spike in temperature. The numerical results have been shifted by -205 cm in space to compensate for the discrepancy in shock position caused by the initial setup.



results in the figure had to be shifted by 10 cm in space to match the analytic shock position. This discrepancy in shock position is due to the initial transient phase as the initial state relaxes to the correct steady-state profile. No flux limiter is used in this calculation because the analytic solution of Lowrie & Edwards (2008) assumes  $\lambda = 1/3$ .

The setup of the Mach 5 shock problem is similar to that of the Mach 2 shock problem. The computational domain in this test is  $-4000 \text{ cm} < x < 2000 \text{ cm}$ . Four refinement levels (five total levels) are used with the finest resolution at  $\Delta x \approx 1.5 \text{ cm}$ . The refinement criteria are the same as in the Mach 2 shock test (Eq. 53). The initial left and right states are  $\rho_L = 5.45887 \times 10^{-13} \text{ g cm}^{-3}$ ,  $T_L = 100 \text{ K}$ ,  $u_L = 5.88588 \times 10^5 \text{ cm s}^{-1}$  and  $\rho_R = 1.96405 \times 10^{-12} \text{ g cm}^{-3}$ ,  $T_R = 855.720 \text{ K}$ ,  $u_R = 1.63592 \times 10^5 \text{ cm s}^{-1}$ , respectively. Again, the system settles down to a steady shock after a brief period of adjustment. Figure 10 shows the results at  $t = 0.04 \text{ s}$ . The results including the narrow spike in temperature are in good agreement with the analytic solution.

### 6.7. Advecting Radiation Pulse

In this test, introduced by Krumholz et al. (2007), we simulate the advection of a radiation pulse by the motion of gas. The initial temperature and density profiles are

$$T = T_0 + (T_1 - T_0) \exp\left(-\frac{x^2}{2w^2}\right), \quad (54)$$

$$\rho = \rho_0 \frac{T_0}{T_1} + \frac{a\mu}{3R} \left( \frac{T_0^4}{T} - T^3 \right), \quad (55)$$

where  $T_0 = 10^7 \text{ K}$ ,  $T_1 = 2 \times 10^7 \text{ K}$ ,  $\rho_0 = 1.2 \text{ g cm}^{-3}$ ,  $w = 24 \text{ cm}$ , the mean molecular weight of the gas is  $\mu = 2.33$ , and  $R$  is the ideal gas constant. Initially the radiation is assumed to be in thermal equilibrium with the gas. We also assume that the radiation pressure is  $E_r/3$  (i.e.,  $\lambda = 1/3$ ). If there were no radiation diffusion, the system would be in an equilibrium with the gas pressure and radiation pressure balancing each other. The interaction coefficients are set proportional to density as  $\kappa_P = \chi_R = 100 \text{ cm}^2 \text{ g}^{-1} \times \rho$ . Because of radiation diffusion, the balance is lost and the gas moves. The purpose of this test is to assess the ability of the code to handle radiation being advected by gas. We will solve the problem numerically in different laboratory frames. Then we can compare the case in which the gas is initially at rest to the case in which the gas initially moves at a constant velocity.

We have performed three runs for comparison with each other and against the results of Krumholz et al. (2007). The computational domain in all three runs is a one-dimensional region of  $-512 \text{ cm} < x < 512 \text{ cm}$  with periodic boundaries. The simulations are stopped at  $t = 4.8 \times 10^{-5} \text{ s}$ . The velocity in the first run is initially zero everywhere, whereas in the second run it is  $10^6 \text{ cm s}^{-1}$  everywhere. The numerical grid in these two runs consists of 512 uniform cells. The third run is a high-resolution calculation of the first case with 4096 uniform cells. Figure 11 shows the density, velocity, and temperature profiles for all three runs. The results of the advected run have been shifted in space for comparison. The profiles from these three runs are almost

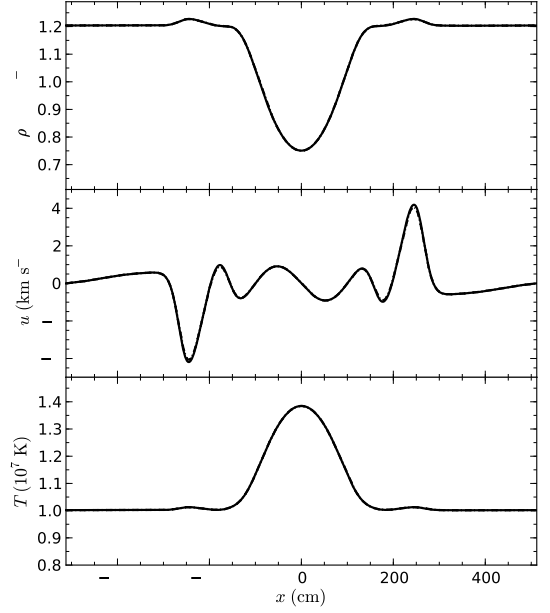


FIG. 11.— Density, velocity and temperature profiles at  $t = 4.8 \times 10^{-5} \text{ s}$  for the test of advecting radiation pulse. The results of three runs are shown. The difference is so small that it is nearly invisible to the eye.

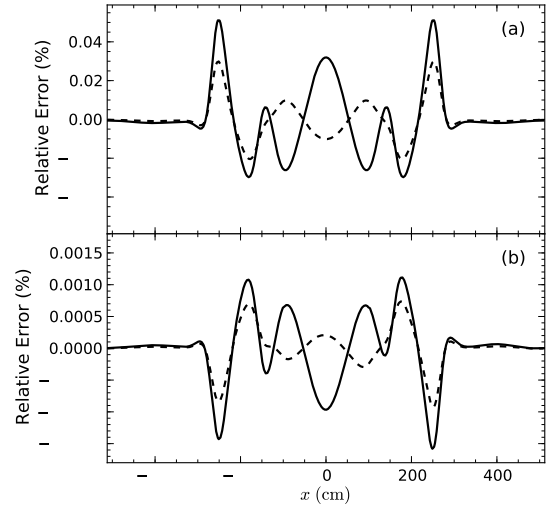


FIG. 12.— Relative errors in density (solid lines) and gas temperature (dashed lines) in the test of advecting radiation pulse. The difference between the low-resolution unadvected and high-resolution unadvected runs is shown in panel (a), whereas the difference between the advected and unadvected runs is shown in panel (b).

identical demonstrating the accuracy of our scheme in this test. We also show the relative difference in Figure 12. The relative error of the low-resolution unadvected run with respect to the high-resolution run is computed as  $(\text{high} - \text{low})/\text{high}$ , and the relative error of the low-resolution advected run with respect to the low-resolution unadvected run is computed as  $(\text{unadvected} - \text{advected})/\text{unadvected}$ . Note that in Figure 12 the results of the high-resolution run have been restricted to the low-resolution grid by averaging for comparison. The figure shows that the difference between the low-resolution and high-resolution runs is less than 0.05%

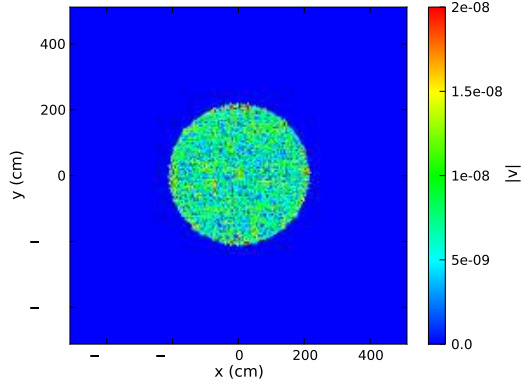


FIG. 13.— Structure of the magnitude of velocity at  $t = 10^{-4}$  s in the static equilibrium test. The unit of velocity is  $\text{cm s}^{-1}$ .

everywhere, and the difference between the advected and unadvected runs is less than 0.0016% everywhere. The relative error in our runs is 1000 times smaller than that of Krumholz et al. (2007). We also note that the asymmetry in the difference between the advected and unadvected cases is expected because our upwinding scheme breaks the symmetry slightly in the advected run. The linear solver can also break the symmetry slightly.

### 6.8. Static Equilibrium

In the test of advecting radiation pulse (§ 6.7), if there were no radiation diffusion, the system would be in a static equilibrium with the gas and radiation pressures balancing each other. In this test, we set both interaction coefficients to a very high value of  $10^{20} \text{ cm}^2 \text{ g}^{-1} \times \rho$ . Thus almost no diffusion can happen. We have performed a calculation on a two-dimensional Cartesian grid of  $-512 \text{ cm} < x < 512 \text{ cm}$  and  $-512 \text{ cm} < y < 512 \text{ cm}$  with 512 uniform cells in each direction. The initial velocity is zero everywhere. For the initial setup, the coordinate  $x$  in Eq. 54 is replaced by  $r = \sqrt{x^2 + y^2}$ . Figure 13 shows the velocity profile at  $t = 10^{-4}$  s. The maximal velocity at that time is  $\sim 4 \times 10^{-8} \text{ cm s}^{-1}$ . Note that the sound speed in this problem is between  $2.5 \times 10^7$  and  $6.1 \times 10^7 \text{ cm s}^{-1}$ . Such a small gas velocity indicates that CASTRO can maintain a perfect static equilibrium in multiple dimensions because of the way radiation pressure and gas pressure are coupled in the Riemann solver.

### 6.9. Radiative Blast Wave: Case 1

In this test, a large amount of energy is deposited into a small region. This results in a spherical blast wave, which is somewhat similar to the Sedov-Taylor blast wave in hydrodynamics, except here we include radiation. There is no analytic solution for this problem.

We use this problem to test the implementation of the geometric factors in CASTRO. We have performed simulations in 1D spherical, 2D cylindrical ( $r$  and  $z$ ), and 3D Cartesian coordinates. Initially, the gas is at rest with a density of  $5 \times 10^{-6} \text{ g cm}^{-3}$ . The initial temperature for both the gas and radiation is set to  $10^3 \text{ K}$ , except for a region inside a sphere with a radius of  $2 \times 10^{12} \text{ cm}$  centered at the origin where temperature is set to  $10^7 \text{ K}$ . The gas is assumed to be ideal with an adiabatic index of  $\gamma = 5/3$  and a mean molecular weight of  $\mu = 1$ . The Planck and Rosseland coefficients are set to  $\kappa_P = 2 \times 10^{-16} \text{ cm}^{-1}$  and

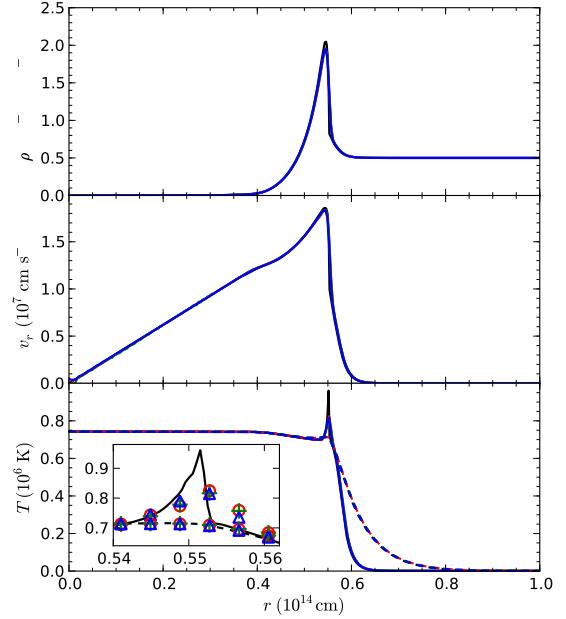


FIG. 14.— Density, radial velocity and temperature profiles at  $t = 10^6$  s for the first radiative blast wave problem. Here, radial velocity is the velocity in spherical radial direction. We show the results of the 1D spherical run (red), 2D cylindrical run (green), 3D Cartesian run (blue), and the high-resolution 1D spherical run (black). The bottom panel shows both the gas temperature (solid lines) and radiation temperature (dashed lines). The inset shows a blow-up of the spike in gas temperature, where the 1D, 2D, 3D, and the high-resolution runs are shown in red circles, green plus signs, blue triangles, and black lines.

$\chi_R = 2 \times 10^{-10} \text{ cm}^{-1}$ . These simulations were run with two refinement levels (three total levels) with a cell size of  $\sim 3.9 \times 10^{11} \text{ cm}$  at the finest level. In these simulations, at each but the finest level a numerical cell is tagged for refinement if it satisfies either  $\rho > 5.01 \times 10^{-6} \text{ g cm}^{-3}$  or  $T > 9 \times 10^6 \text{ K}$ . Furthermore, the intermediate level is slightly larger than the finest level due to proper nesting (see Paper I). The computational domain for the 1D spherical run is  $0 < r < 10^{14} \text{ cm}$ . The computational domain for the 2D cylindrical run is  $0 < r < 10^{14} \text{ cm}$  and  $-10^{14} \text{ cm} < z < 10^{14} \text{ cm}$ . For the 3D run, the computational domain is  $(-10^{14}, 10^{14}) \text{ cm}$  in each direction. A CFL number of 0.6 is used for these simulations, and the initial time step is shrunk by a factor of 100 to allow the point explosion to develop. Note that we have chosen these initial conditions so that different regimes can be explored by the test. The blast wave starts with almost all the energy in radiation. At the end of the simulations, approximately one third of the energy is in the gas, and the gas pressure just behind the shock is about twice the radiation pressure. Furthermore, the gas and radiation are not in equilibrium near the shock due to the low Planck mean opacity.

Since there is no analytic solution for the problem, we have also run a high-resolution 1D simulation with a cell size of  $\sim 4.9 \times 10^{10} \text{ cm}$  for comparison. Figure 14 shows the radial profiles of density, radial velocity, gas temperature, and radiation temperature at  $t = 10^6$  s for these runs. The radial profiles of the 2D and 3D results are computed by mapping each cell into its corresponding radial bin and averaging. The width of the bins is chosen to be the cell size at the finest refinement level. The

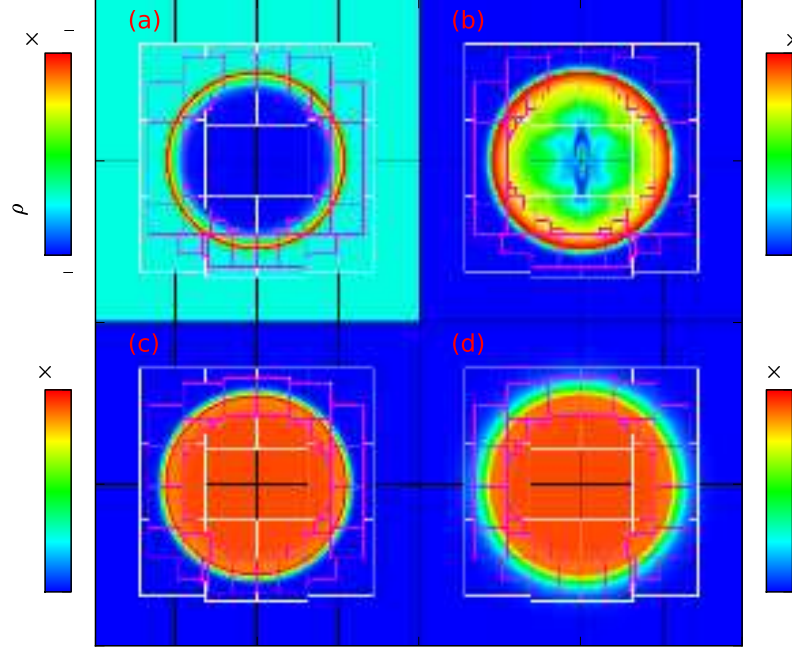


FIG. 15.— Snapshot of the 2D cylindrical simulation at  $t = 10^6$  s for the first radiative blast wave test. Profiles of density, radial velocity, gas temperature, and radiation temperature are shown in panels (a), (b), (c), and (d), respectively. Here radial velocity is the velocity in spherical radial direction, not the cylindrical radial direction. The physical domain in each dimension is  $(-10^{14}, 10^{14})$  cm for each panel. Also shown is the grid structure of the adaptive mesh. Levels 0, 1, and 2 are shown in black, white, and purple, respectively.

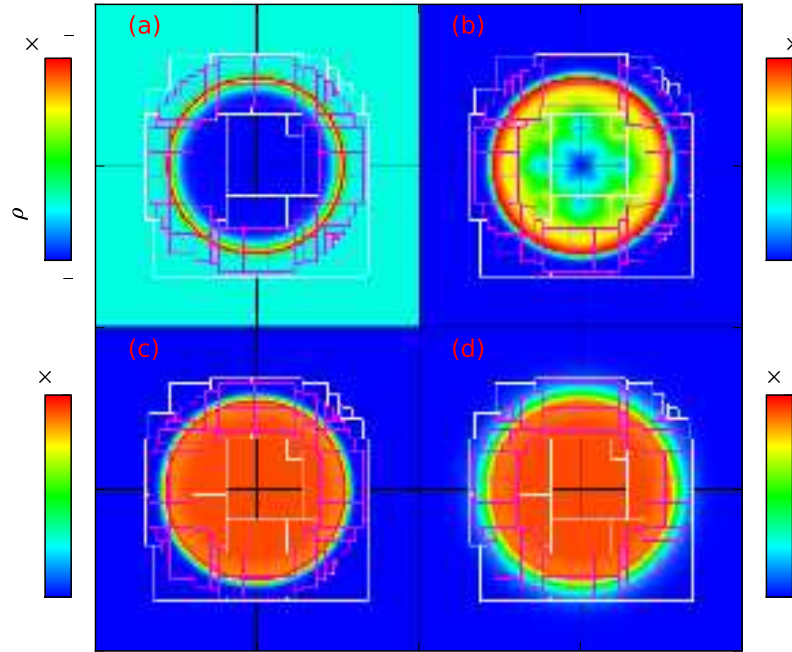


FIG. 16.— 2D slice at  $z = 0$  of the 3D Cartesian simulation at  $t = 10^6$  s for the first radiative blast wave test. Profiles of density, radial velocity, gas temperature, and radiation temperature are shown in panels (a), (b), (c), and (d), respectively. Here radial velocity is the velocity in spherical radial direction. The physical domain in each dimension is  $(-10^{14}, 10^{14})$  cm for each panel. Also shown is the grid structure of the adaptive mesh. Levels 0, 1, and 2 are shown in black, white, and purple, respectively.

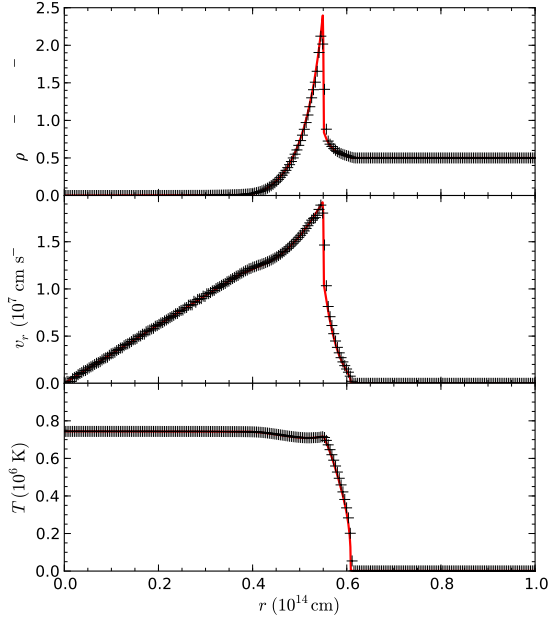


FIG. 17.— Density, radial velocity and temperature profiles at  $t = 10^6$  s for the second radiative blast wave problem. The results of the 2D cylindrical simulation are shown in symbols (*black plus signs*). We do not show the radiation temperature because it is almost identical to the gas temperature. Here, radial velocity is the velocity in spherical radial direction, not the cylindrical radial direction. Also shown are the results of the high-resolution 1D spherical simulation (*red solid lines*).

results from runs in three different coordinates are in excellent agreement with each other, and they agree with those of the high-resolution simulation. A snapshot of the structure of density, radial velocity, gas temperature, and radiation temperature at  $t = 10^6$  s for the 2D simulation is shown in Figure 15. A 2D slice at  $z = 0$  of the 3D simulation is shown in Figure 16. The multi-D results show good agreement with each other, and they are spherically symmetric except for some minor asymmetry in velocity at the low density region near the center. This asymmetry is, in part, due to the representation of the initial hot sphere in non-spherical coordinates. Another source of the asymmetry in the 2D run is the coordinate singularity at the longitudinal axis ( $r = 0$ ) of the cylindrical coordinates. It should also be noted that the AMR gridding algorithm in CASTRO does not necessarily preserve symmetry even if the layout of the numerical cells that are tagged for refinement is symmetric. Last, but perhaps not least, the linear solver does not preserve perfect symmetry either.

#### 6.10. Radiative Blast Wave: Case 2

This test is similar to the first radiative blast wave test (§ 6.9) except that the Planck mean coefficient is set to  $\kappa_P = 2 \times 10^{-7} \text{ cm}^{-1}$ . Thus the ratio of the two coefficients is  $\kappa_P/\chi_R = 1000$ . Because of such a large ratio of the two mean opacities, the Lorentz transformation term has to be treated implicitly in this test, otherwise the time step would have to be much smaller for stability reasons.

We have performed a 2D cylindrical simulation with two refinement levels (three total levels) and a cell size of  $\sim 3.9 \times 10^{11} \text{ cm}$  at the finest level on a computational domain of  $0 < r < 10^{14} \text{ cm}$  and  $-10^{14} \text{ cm} < z < 10^{14} \text{ cm}$ . In this 2D AMR run, a numerical cell is tagged for re-

finement if it satisfies either  $\rho > 5.01 \times 10^{-6} \text{ g cm}^{-3}$  or  $T > 9 \times 10^6 \text{ K}$ . We have also performed a high-resolution 1D simulation with a cell size of  $\sim 4.9 \times 10^{10} \text{ cm}$  for comparison. A CFL number of 0.6 is used for the simulations, and the initial time step is shrunk by a factor of 100 to allow the point explosion to develop. Figure 17 shows the radial profiles of density, radial velocity, gas temperature and radiation temperature at  $t = 10^6$  s for the two runs. The results show that CASTRO can handle a very large Lorentz transformation term implicitly without having to decrease the time step.

## 7. SUMMARY

We have developed a new radiation hydrodynamics solver in our compressible astrophysics code, CASTRO. We use the mixed-frame approach and adopt the FLD assumption. The solver uses a second-order explicit Godunov method for the hyperbolic part of the system and a first-order backward Euler method for the parabolic part. We have also presented the mathematical characteristics of the hyperbolic subsystem. The eigenvalues and eigenvectors of the system we have obtained are used to construct the Riemann solver in our Godunov scheme, and could also be useful for other characteristic-based schemes.

We have demonstrated the capability of CASTRO to address a wide range of radiation hydrodynamics problems by extensive testing. There are a number of other radiation hydrodynamics codes. Some recent examples include ZEUS-2D (Turner & Stone 2001), ZEUS-MP (Hayes et al. 2006), Orion (Krumholz et al. 2007), HERACLES (González et al. 2007), V2D (Swesty & Myra 2009), Nike (Sekora & Stone 2010), RAMSES (Commerçon et al. 2011), and CRASH (van der Holst et al. 2011). Here we compare CASTRO with these other codes.

- A major advantage of CASTRO is its efficiency due to the use of AMR and combined with good scaling behavior on up to 32768 cores. Among the other radiation hydrodynamics codes listed above, only Orion, RAMSES, and CRASH are three-dimensional AMR codes. ZEUS-MP and HERACLES are three-dimensional codes without AMR. ZEUS-2D and V2D are two-dimensional codes. Nike is a one-dimensional code.
- A unique strength of our code is that the hyperbolic solver in CASTRO is an unsplit version of the PPM method that avoids spurious noise caused by dimensional splitting (see Paper I for an example of the advantage of unsplit methods over dimensional splitting methods). For the hyperbolic part, Orion, HERACLES, Nike, RAMSES, and CRASH use Godunov methods, whereas ZEUS-2D, ZEUS-MP, and V2D use the ZEUS type of algorithms. The latter are faster but somewhat less accurate than high-order Godunov methods. However, for a radiation hydrodynamics code, the cost of the hyperbolic solver is no longer a concern, because the implicit parabolic solver is more expensive. For multigroup radiation hydrodynamics, the implicit parabolic solver would be even more expensive and the hyperbolic solver would be essentially free.



- Our scheme is very robust even for dynamic diffusion. By solving the entire hyperbolic subsystem in one Riemann solver, the scheme can avoid the operator splitting errors that appear in many other numerical schemes (see e.g., Krumholz et al. 2007). We note that RAMSES and CRASH also couple both radiation and matter in their Riemann solvers.
- CASTRO is based on a mixed-frame formulation, as is Orion and Nike. A main advantage of the mixed-frame approach is that it conserves the total energy, whereas its main disadvantage is that it is of limited use for line transport. However, line transport cannot be treated by a gray radiation solver regardless of its choice of frame.
- CASTRO, ZEUS-2D, ZEUS-MP, Orion, V2D, RAMSES, and CRASH have adopted the FLD approach, whereas HERACLES and Nike are based on the two-moment approach. Thus CASTRO carries the limitations of FLD, such as poor accuracy for optically-thin flows. However, FLD is computationally cheaper than the two-moment approach. Moreover, for multigroup radiation, FLD is much less memory intensive than the two-moment approach.

The current implementation uses a gray approximation based on the frequency-integrated formulation of the radiation hydrodynamics equations. We note that numerical codes exist for multigroup flux-limited radiation hy-

drodynamics (e.g., Burrows et al. 2007; Swesty & Myra 2009; van der Holst et al. 2011). It is straightforward to extend our scheme to multigroup radiation because the radiation pressure in the hyperbolic subsystem is still a frequency-integrated quantity for multigroup radiation. A multigroup neutrino-radiation hydrodynamics solver is also currently under development.

Further details on CASTRO can be found in the CASTRO User Guide (CASTRO User Guide 2011).

The authors would like to thank Eric Myra, Doug Swesty and Michael Zingale at Stony Brook University for a number of helpful discussions about radiation hydrodynamics. The work at LBNL was supported by the Office of High Energy Physics and the Office of Advanced Scientific Computing Research of the U.S. Department of Energy under contract No. DE-AC02-05CH11231. The work performed at LLNL was supported by the SciDAC program of the U.S. Department of Energy under the auspices of contract No. DE-AC52-07NA27344. Adam Burrows was supported by the SciDAC program of DOE under grant number DE-FG02-08ER41544, the NSF under subaward no. ND201387 to the Joint Institute for Nuclear Astrophysics, and the NSF PetaApps program, under award OCI-0905046 via a subaward no. 44592 from Louisiana State University to Princeton University. This research used resources of the National Energy Research Scientific Computing Center, which is supported by the Office of Science of the U.S. Department of Energy under Contract No. DE-AC02-05CH11231.

## REFERENCES

- Alme, M. L., & Wilson, J. R. 1973, *ApJ*, 186, 1015
- Almgren, A. S. et al. 2010, *ApJ*, 715, 1221
- Bell, J. B., Colella, P., & Trangenstein, J. A. 1989, *Journal of Computational Physics*, 82, 362
- Berger, M. J., & Colella, P. 1989, *Journal of Computational Physics*, 82, 64
- Burrows, A., Livne, E., Dessart, L., Ott, C. D., & Murphy, J. 2007, *ApJ*, 655, 416
- Castor, J. I. 1972, *ApJ*, 178, 779
- . 2004, *Radiation Hydrodynamics* (Cambridge, UK: Cambridge University Press)
- CASTRO User Guide. 2011, <https://ccse.lbl.gov/Research/CASTRO/CastroUserGuide.pdf>
- Colella, P., Glaz, H. M., & Ferguson, R. E. 1997, unpublished manuscript
- Commerçon, B., Teyssier, R., Audit, E., Hennebelle, P., & Chabrier, G. 2011, *A&A*, 529, A35+
- Falgout, R., & Yang, U. 2002, in *Lecture Notes in Computer Science*, Vol. 2331, *Computational Science ICCS 2002*, ed. P. Sloot, A. Hoekstra, C. Tan, & J. Dongarra (Springer Berlin / Heidelberg), 632–641
- Fryxell, B. et al. 2000, *ApJS*, 131, 273
- González, M., Audit, E., & Huynh, P. 2007, *A&A*, 464, 429
- Hayes, J. C., & Norman, M. L. 2003, *ApJS*, 147, 197
- Hayes, J. C., Norman, M. L., Fiedler, R. A., Bordner, J. O., Li, P. S., Clark, S. E., ud-Doula, A., & Mac Low, M.-M. 2006, *ApJS*, 165, 188
- Howell, L. H., & Greenough, J. A. 2003, *Journal of Computational Physics*, 184, 53
- Hubeny, I., & Burrows, A. 2007, *ApJ*, 659, 1458
- hydre Code Project. 2011, <http://www.llnl.gov/CASC/hybre/>
- Krumholz, M. R., Klein, R. I., McKee, C. F., & Bolstad, J. 2007, *APJ*, 667, 626
- LeVeque, R. J. 2002, *Finite-Volume Methods for Hyperbolic Problems* (Cambridge University Press)
- Levermore, C. D. 1984, *Journal of Quantitative Spectroscopy and Radiative Transfer*, 31, 149
- Levermore, C. D., & Pomraning, G. C. 1981, *ApJ*, 248, 321
- Lowrie, R. B., & Edwards, J. D. 2008, *Shock Waves*, 18, 129
- Lowrie, R. B., Morel, J. E., & Hittinger, J. A. 1999, *ApJ*, 521, 432
- McCormick, S. F. 1989, *Multilevel Adaptive Methods for Partial Differential Equations* (Philadelphia: SIAM)
- Mihalas, D., & Klein, R. I. 1982, *Journal of Computational Physics*, 46, 97
- Mihalas, D., & Mihalas, B. W. 1999, *Foundations of radiation hydrodynamics* (New York: Dover)
- Miller, G. H., & Colella, P. 2002, *Journal of Computational Physics*, 183, 26
- Pomraning, G. C. 1973, *The equations of radiation hydrodynamics* (Oxford: Pergamon Press)
- . 1979, *J. Quant. Spec. Radiat. Transf.*, 21, 249
- Sekora, M. D., & Stone, J. M. 2010, *Journal of Computational Physics*, 229, 6819
- Su, B., & Olson, G. L. 1996, *J. Quant. Spec. Radiat. Transf.*, 56, 337
- Swesty, F. D., & Myra, E. S. 2009, *ApJS*, 181, 1
- Turner, N. J., & Stone, J. M. 2001, *ApJS*, 135, 95
- van der Holst, B. et al. 2011, *ApJS*, 194, 23
- Zel'Dovich, Y. B., & Raizer, Y. P. 1967, *Physics of shock waves and high-temperature hydrodynamic phenomena* (New York: Academic Press)